

**VŠB - Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra kybernetiky a biomedicínského inženýrství**

**Použití technologie IQRF pro monitorování fyzikálních veličin**  
**Using IQRF Technology for Physical Quantities Monitoring**

## Zadání diplomové práce

Student:

**Bc. Zdeněk Macháč**

Studijní program:

N2649 Elektrotechnika

Studijní obor:

2601T004 Měřicí a řídicí technika

Téma:

**Použití technologie IQRF pro monitorování fyzikálních veličin**  
**Using IQRF Technology for Physical Quantities Monitoring**

Zásady pro vypracování:

1. Nastudování problematiky orientace v prostoru.
2. Nastudování hardwarových prostředků pro realizaci orientace v prostoru s bezdrátovým přenosem.
3. Návrh a realizace konstrukce měřicí části s bezdrátovým přenosem.
4. Návrh a realizace obslužného software pro zobrazení a archivaci dat na PC.
5. Testování vytvořeného prototypu.
6. Zhodnocení dosažených výsledků.

Seznam doporučené odborné literatury:

- [1] SEFLOVA, Petra et al. IQRF wireless technology utilizing IQMESH protocol. In: *35th International Conference on Telecommunications and Signal Processing, TSP 2012*. Prague, Jul. 2012, p. 101-104. ISBN 978-1-4673-1117-5. doi: 10.1109/TSP.2012.6256261.
- [2] MICRORISC s.r.o. *Complete technology for MESH networks*. [online]. [cit.2014-10-16]. c2013. Dostupné z: <http://iqrf.org/weben/index.php>.
- [3] VRBA, Kamil et al. *Konstrukce elektronických zařízení*. Brno: Vysoké učení technické v Brně, 2013. s. 251. ISBN 978-80-214-4719-6. Dostupné také z: [http://homen.vsb.cz/~vac174/vyuka/KZPE/Studijni\\_opory/Konstrukce\\_elektronickych\\_zarizeni.pdf](http://homen.vsb.cz/~vac174/vyuka/KZPE/Studijni_opory/Konstrukce_elektronickych_zarizeni.pdf)
- [4] GANG, Sun et al. Implementing quaternion based AHRS on a MEMS multisensor hardware platform. In: *International Global Navigation Satellite Systems Society IGNSS Symposium 2013*. Outrigger Gold Coast, Qld Australia, Jul. 2013.
- [5] MARCARD, Timo Von. *Design and implementation of an attitude estimation system to control orthopedic components*. Göteborg, Sweden: CHALMERS UNIVERSITY OF TECHNOLOGY, 2010. Master's Thesis EX032/2010. Dostupné také z: <http://publications.lib.chalmers.se/records/fulltext/125985.pdf>

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Martin Pieš, Ph.D.**

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015

doc. Ing. Jiří Koziorek, Ph.D.  
vedoucí katedry

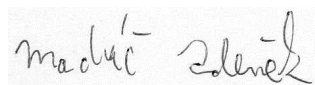


prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## Prohlášení

Prohlašuji, že jsem tuto diplomovou práci na téma „Použití technologie IQRF pro monitorování fyzikálních veličin“ vypracoval samostatně s použitím uvedené odborné literatury a pramenů. Tato práce byla vypracována pod vedením pana Ing. Martina Pieše, Ph.D.

Dále prohlašuji, že jsem uvedl všechny literární prameny a publikace, ze kterých jsem čerpal.

A handwritten signature in black ink, reading "Macháč Zdeněk". The signature is written in a cursive style.

Zdeněk Macháč

Datum odevzdání diplomové práce: 7.5.2015

## **Poděkování**

Chtěl bych poděkovat své vedoucí diplomové práce Ing. Martinu Piešovi, Ph.D. za odborné vedení, za pomoc a rady při zpracování této práce. Dále bych chtěl poděkovat Bc. Jiřímu Pospíšilovi za pomoc při výrobě DPS a cenné rady.

## **Abstrakt**

Tato práce se zabývá návrhem a konstrukcí bezdrátového vysílače a přijímače IQRF modulů pro měření hodnot potřebných pro orientaci v prostoru a jejich následná vizualizace.

Řešení popsané v této práci vede k funkčnímu návrhu a konstrukci obou modulů a následné vizualizaci v počítači pomocí programovacího prostředí LabVIEW. Hlavní podstatou vizualizace je AHRS algoritmus vedoucí k přesnějším výsledkům rotace objektu v prostoru.

Podařilo se vytvořit podprogram pro 3D rotaci objektu a 3D graf kreslící trajektorii pohybu objektu v prostoru.

Výsledky této práce ukazují na nevhodnost IQRF modulů na provoz větších vzdáleností. Např. použití pro quadcoptery, dálkově ovládané vozidla apod. Jedná se ovšem o vynikající prostředek pro ovládání chytrého domu, rozhlasu na pouličních lampách apod.

## **Klíčová slova**

IQRF, AHRS, AltIMU, gyroskop, akcelerometr, magnetometr, I<sup>2</sup>C

## **Abstract**

This work describes the design and construction of a wireless transmitter and receiver modules IQRF for measurement values needed for orientation in space and their visualization.

The solution described in this work leads to the functional design and construction of both modules and visualization in computer with LabVIEW programming environment. The main element of visualization is the AHRS algorithm, which leads to more accurate results rotation of the object in space

We managed to create a subroutine for 3D rotation of the object's and 3D graph plotting trajectory of the object in space.

The results of this work demonstrate the unsuitability IQRF modules operate at greater distances. For example, use for quadcopter, remotely operated vehicles etc. This is an excellent thing to control the smart house, radio on street lamps, etc.

## **Keywords**

IQRF, AHRS, AltIMU, gyroscope, accelerometer, magnetometer, I<sup>2</sup>C

## Seznam použitých zkratk a symbolů

IQRF	Inteligentní rádiový modul
PC	Personal computer (osobní počítač)
AHRS	Attitude and heading reference system (poloha a kurs referenčního systému)
VISA	Virtual Instrument Software Architecture (Komunikační protokol)
Wi-fi	Bezdrátová počítačová síť
EEPROM	Electrically Erasable Programmable Read-Only Memory (elektricky mazatelná paměť, uchování dat je nezávislé na napájení)
VL	Low voltage (napětí s nízkou úrovní)
VH	High voltage (napětí s vysokou úrovní)
USB	Universal Serial Bus (univerzální sériová sběrnice, nejčastěji používaná komunikace v PC)
OS	Operační systém
RF	Rádiová komunikace
I <sup>2</sup> C	Inter-Integrated Circuit (počítačová sériová sběrnice pro nízkorychlostní periferie)
UART	Komunikace sériové linky (taktéž známá jako RS-232)
Li-Pol	Lithium-polymerový akumulátor
SMD	Součástka pro povrchovou montáž plošných spojů
DPS	Deska plošných spojů
LabVIEW	Programovací prostředí od společnosti National instruments

## Obsah

1	Úvod.....	1
2	Nastudování problematiky orientace v prostoru .....	2
3	Nastudování hardwarových prostředků pro realizaci orientace v prostoru s bezdrátovým přenosem .....	5
3.1	Gyroskop.....	5
3.2	Akcelerometr .....	6
3.3	Magnetometr .....	7
3.4	Barometr .....	8
3.5	GPS modul .....	8
3.6	Ostatní .....	8
3.6.1	Ultrazvukový senzor .....	8
3.6.2	Infračervené senzory .....	8
3.6.3	Optovláknové senzory .....	8
4	Návrh a realizace konstrukce měřicí části s bezdrátovým přenosem .....	9
4.1	Komunikační modul IQRF .....	9
4.2	Modul s čidly AltIMU-10 v4 .....	11
4.3	Schéma propojení modulů .....	13
4.4	DPS a krabíčka.....	14
4.4.1	DPS .....	14
4.4.2	Krabíčka.....	15
4.5	Akumulátor a nabíječka .....	16
4.6	Komunikace mezi moduly .....	17
4.7	Software měřicí částí .....	18
4.7.1	První část.....	19
4.7.2	Druhá část .....	19
4.7.3	Nastavení registrů čidel .....	19
4.8	Operační systém modulu TR-56D .....	20
5	Návrh a realizace obslužného software pro zobrazení a archivaci dat na PC.....	21
5.1	Popis programovacího prostředí .....	21
5.1.1	Front Panel .....	21
5.1.2	Block diagram .....	22
5.1.3	Vytvoření .exe souboru a instalačního souboru .....	22
5.2	Sběr dat .....	23
5.2.1	Komunikace UART .....	23
5.2.2	Nastavení UARTu.....	24
5.3	Vizualizace.....	25
5.3.1	Program vizualizace.....	25
5.3.2	Popis jednotlivých funkcí .....	27
5.3.3	Kalmanův filtr .....	31
5.3.4	Výpočet rotace pomocí kvaternionů .....	34
5.4	Čelní panel vizualizace .....	35
5.4.1	Záložka Nastavení.....	35
5.4.2	Záložka Akcelerometr, Gyroskop a Magnetometr .....	36
5.4.3	Záložka 3D graf .....	37
5.4.4	Záložka Trajektorie.....	39



6	Testování vytvořeného prototypu .....	41
6.1	Dosah měřicího modulu .....	41
6.2	Rotace kolem osy Z .....	43
6.2.1	Akcelerometr .....	43
6.2.2	Magnetometr .....	44
6.2.3	Gyroskop .....	45
6.2.4	3D graf rotace .....	46
6.2.5	Trajektorie .....	47
7	Zhodnocení dosažených výsledků .....	49
8	Seznam literatury .....	51
9	Seznam příloh .....	54
9.1	Přílohy pouze na CD .....	54

## 1 Úvod

V zadání je napsáno **použití technologie IQRF pro monitorování fyzikálních veličen**, ale jde pouze o obecný název. Tato diplomová práce se zabývá měřením pohybových veličin a z nich se následně vypočítává rotace objektu a trajektorie pohybu měřicího modulu v prostoru. Trasa je měřena ve 3D pomocí tříosého gyroskopu, tříosého akcelerometru a tříosého magnetometru. Obsluhu těchto čidel obstarává IQRF modul TR-56D, který je popsán v kapitole 3.1, a který umí komunikovat bezdrátově s jím kompatibilním modulem na vysílací frekvenci 868MHz.

**Celá práce se dá rozdělit na dvě samostatné části:**

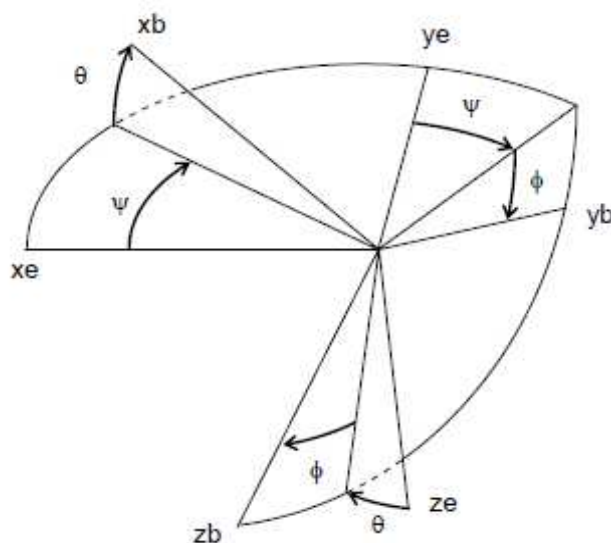
- **Vysílací část (měřicí modul)** - naměří pomocí čidel aktuální hodnoty zrychlení, zemského magnetického pole a rotaci gyroskopu a pomocí modulu TR-56D odešle data do přijímací části. Měřicí část pouze data naměří a odešle. K zpracování dojde až ve vizualizaci na PC. Je to tak navrženo proto, aby nebyly zbytečně zatěžovány měřicí moduly a mohly soustředit veškerou výkonovou kapacitu na měření.

- **Přijímací část** - Údaje z těchto měřicích modulů (vysílacích částí) jsou zpracovány v PC pomocí aplikace napsanou v programovacím prostředí NI LabVIEW. Zde jsou data zpracovány pomocí VISA protokolu a dále jsou data filtrována pomocí Kalmanova filtru. Po odfiltrování dat přichází na řadu AHRS algoritmus (výpočet kvaternionů a následný výpočet rotace a trajektorie). Pro ukázkou jsou zobrazeny jak data surová tak data odfiltrovaná a to pomocí grafů, kde je zobrazena jednotka v závislosti na počtu vzorků.

Součástí vizualizace je uložení naměřených hodnot do textového souboru. Po stisku tlačítka na uložení dat se objeví ohlášení, že ne/byly data uloženy.

## 2 Nastudování problematiky orientace v prostoru

Pro pohyb v prostoru je potřeba si definovat souřadnicový systém. Souřadnicový systém je definovaný buďto osami fixně danými vůči Zemi nebo jsou to souřadnicové systémy využívající vázané k dané konstrukci zařízení. Obrázek 1 ukazuje oba tyto souřadnicové systémy, kde symboly  $x_e$ ,  $y_e$ ,  $z_e$  jsou svázány s vůči Zemi a symboly  $x_b$ ,  $y_b$ ,  $z_b$  patří k dané konstrukci zařízení. [16]



Obrázek 1: Znázornění souřadnicových systémů [16]

Prostor je standardně definován třemi osami - horizontální, vertikální a předozadní (XYZ). K těmto osám patří i související natočení těchto os standardně označovány Pitch, Roll, Yaw. Pro natočení podle obrázku 1 se používají také symboly  $\Theta$   $\Phi$   $\Psi$ . Kde získat vhodná data k výpočtu rotací? Nejvhodnějším kandidátem je bezpochyby gyroskop, který dává přímo hodnotu natočení dané osy. Pro přesnější výpočet a odstranění nedostatku gyroskopu se často přidávají další čidla a to konkrétně akcelerometr a magnetometr. Magnetometr při správném nastavení ukazuje, kterým směrem je sever a tudíž se z jeho dat dá vypočítat také rotace. Problémem tohoto čidla jsou ovšem velké rušivé elementy jako jsou elektromagnetické rušení okolí (telefonní sítě, wi-fi, elektrická síť apod.) nebo zkreslení okolními kovovými předměty. Částečně se tyto elementy dají odstranit, ale nikdy ne dokonale. Posledním důležitým čidlem je akcelerometr, který udává nejen dynamické zrychlení v jednotlivých osách, ale také statické gravitační zrychlení Země. Pomocí gravitace Země jsme schopni určit, jakým směrem jsme objektem rotovali. Nevýhodou akcelerometru je ovšem fakt, že měří jak statické tak dynamické zrychlení, a proto se samostatně k výpočtu rotace nehodí.

Dohromady jsou ale tyto tři čidla mocný nástroj pro výpočet jak trajektorie, tak rotace kolem všech tří os.

Jak bylo již výše napsáno, rotace os lze spočítat samostatně pomocí jednotlivých čidel. Výpočet pomocí akcelerometru podle rovnic (1.1) ukazuje, jak se rotace mění. Zde ale dochází k velkým chybám při jiném pohybu než-li rotačním.

$$\begin{aligned} pitch(\theta) &= \frac{180}{\pi} \cdot \arctan\left(\frac{AccX}{AccZ}\right) [^\circ] \\ roll(\phi) &= \frac{180}{\pi} \cdot \arctan\left(\frac{AccY}{AccZ}\right) [^\circ] \\ yaw(\psi) &= \frac{180}{\pi} \cdot \arctan\left(\frac{AccX}{AccY}\right) [^\circ] \end{aligned} \quad (1.1)$$

kde  $\theta$  natočení vzhledem k ose  $y_b$

$\phi$  natočení vzhledem k ose  $x_b$

$\psi$  natočení vzhledem k ose  $z_b$

$AccX, AccY, AccZ$  je údaj z akcelerometru (současná - minulá hodnota)

Další možností je získání dat pomocí gyroskopu a vypočítat rotace pomocí těchto dat podle rovnic (1.2.).

$$\begin{aligned} \theta &= \theta + GyroY \cdot \Delta t [^\circ] \\ \phi &= \phi + GyroX \cdot \Delta t [^\circ] \\ \psi &= \psi + GyroZ \cdot \Delta t [^\circ] \end{aligned} \quad (1.2)$$

kde  $GyroX, GyroY, GyroZ$  je údaj z gyroskopu

$\Delta t$  čas mezi vzorky z gyroskopu

Výpočet pomocí magnetometru je obdobný jako u akcelerometru. Hlídají se předchozí hodnoty a ty se pak porovnávají s dalšími daty.

$$\begin{aligned} pitch(\theta) &= \frac{180}{\pi} \cdot \arctan\left(\frac{MagY}{MagZ}\right) [^\circ] \\ roll(\phi) &= \frac{180}{\pi} \cdot \arctan\left(\frac{MagX}{MagZ}\right) [^\circ] \\ yaw(\psi) &= \frac{180}{\pi} \cdot \arctan\left(\frac{MagX}{MagY}\right) [^\circ] \end{aligned} \quad (1.3)$$

kde  $MagX, MagY, MagZ$  je údaj z magnetometru

Výše uvedené vztahy popisují rotace os pouze pomocí jednotlivých čidel. V reálné aplikaci je potřeba výsledky seskupit v jeden. Bylo by sice možné jednotlivé rotace sečíst a vydělit třemi (např.  $(aX+gX+mX)/3$  kde  $aX$  je údaj z akcelerometru osa

X atd.), ale tento způsob není moc vhodný. Spřažení těchto tří čidel se zabývá AHRS algoritmus, který popisuje jak jednotlivá čidla přispívají k rotaci dané osy.

Detailní popis AHRS algoritmu je popsán v kapitole 4.3.1, nicméně výpočet podle tohoto algoritmu vede k lepším výsledkům rotace a výpočtu trajektorie. AHRS (Attitude and heading reference systém), volně přeloženo jako poloha a kurs referenčního systému, se skládá z tříosých senzorů, které poskytují informace o natočení objektu. AHRS algoritmus řeší, jak přepočítat data z jednotlivých čidel na reference a následně se z těchto referencí vypočítávají kvaterniony. Pomocí těchto kvaternionů se dále počítá rotace os a trajektorie. Jinou variantou představuje výpočet rotace pomocí Eulerových úhlů. Tato varianta je popsána výše pomocí vztahů (1.1) až (1.3). Tento popis je pro rotaci složitý na výpočet, protože se každý následující krok musí přepočítávat pomocí těchto úhlů na skutečné osy. Výpočet pomocí kvaternionů pouze tyto úhly sčítá a proto je výpočet jednodušší.

Velkou výhodou moderní doby je, že se čidla dají koupit zapojené na jednom modulu (na jedné DPS) a není tak nutné vymýšlet strategii propojí těchto čidel a komunikaci s nimi. Příkladem je modul AltIMU blíže popsán v kapitole 3.2, který má na jedné DPS tříosé čidla magnetometr, akcelerometr a gyroskop. Nejnovější verze má ještě navíc barometr pro větší uživatelský rozlet.

### 3 Nastudování hardwarových prostředků pro realizaci orientace v prostoru s bezdrátovým přenosem

Pro orientaci v prostoru je možné využít senzory jako je gyroskop, akcelerometr, magnetometr, barometr a další jejich popis je uveden níže. Nejdříve je potřeba si rozmyslet na jaký účel dané zařízení bude a kde se bude pohybovat. Jde-li o venkovní prostor je možné využít GPS a naopak ve vnitřních prostorech toto čidlo nevyužijeme.

#### 3.1 Gyroskop

Gyroskop je z pohledu historie novinka. Jeho princip byl objeven na začátku 18. století a proslavil jej Léon Foucault, který na něm demonstroval rotaci zemské osy.

Tento senzor má mnoho podob, ale hlavní podstatou je setrvačnický otáčející se kolem osy. Setrvačnický klade odpor při změně osy rotace díky svému momentu setrvačnosti. Matematicky je tento moment zapsán podle rovnice (2.1). [7]

$$I = M \cdot R^2 \quad (2.1)$$

kde  $I$  je moment setrvačnosti  
 $M$  je hmotnost tělesa  
 $R$  je vzdálenost tělesa od osy

Hlavní využití je v detekci a měření rotačního pohybu, zpřesnění pozice GPS systémů, měření setrvačnosti a náklonu atd. V technologii MEMS se využívá Coriolisovy síly, což je virtuální síla působící na objekt o hmotnosti  $m$ , který se pohybuje rychlostí  $v$  v soustavě rotující kolem osy úhlovou rychlostí  $\omega$  podle vzorce (2.2).

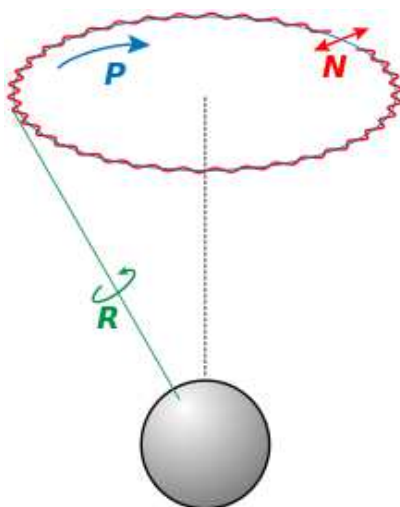
$$F_c = 2 \cdot m \cdot v \times \omega [N] \quad (2.2)$$

kde  $F_c$  je Coriolisova síla  
 $m$  je hmotnost  
 $v$  je rychlost tělesa  
 $\omega$  je úhlová rychlost

#### Typy gyroskopů:

- Rotační
- Vibrační
- Optické

Rotační pohyb gyroskopu je doprovázen nedokonalostmi jako je precese a nutace, které jsou zobrazeny na obrázku 2.



Obrázek 2: Precese a nutace rotujícího tělesa [6]

- Precesí je nazýván posuv osy rotace kolem nehybného směru v prostoru. Tímto jevem prochází i naše planeta Země. Jde o působení jiných těles (Měsíce a Slunce) na Zemi. [7]
- Nutace je zapříčiněná nesymetrickým tělesem vůči ose otáčení. Jde o kolísavý nebo kývavý pohyb osy otáčení tělesa.[6]

Výstupem digitálního gyroskopu je  $^{\circ}/s$ , respektive  $rad/s$ , ten ale může být i analogový a pak je potřeba výstupní hodnotu převést pomocí A/D převodníku a následně přepočítat na  $^{\circ}/s$ . [8][16]

### 3.2 Akcelerometr

Akcelerometr je senzor sloužící pro měření zrychlení a to jak statického, tak i dynamického. Statické zrychlení je reprezentováno tíhovým (gravitačním) zrychlením Země a dynamické zrychlení je pak změna rychlosti v daném směru. Stejně jako gyroskop může akcelerometr měřit více os najednou. Těmito čidly se dá měřit náklon, zrychlení nebo otřesy (vibrace).

#### Typy akcelerometrů:

- Tenzometrické
- Piezoelektrické
- Kapacitní
- Tepelné

Princip měření se liší v závislosti na technologii, ale jsou si vzájemně podobné. Základním principem je např. proměnná kapacita vzduchového kondensátoru. Dvě desky jsou statické a mezi nimi se pohybuje třetí, která pomocí pružin převádí sílu  $F$  (respektive zrychlení  $a$  násobené hmotností této elektrody  $m$ ) na kapacitu  $C$ . Na obrázku

3 je znázorněné zapojení tohoto senzoru. Nelinearita kapacity v závislosti na vzdálenosti elektrod se vypočítá podle vztahu (2.3). [8][9]

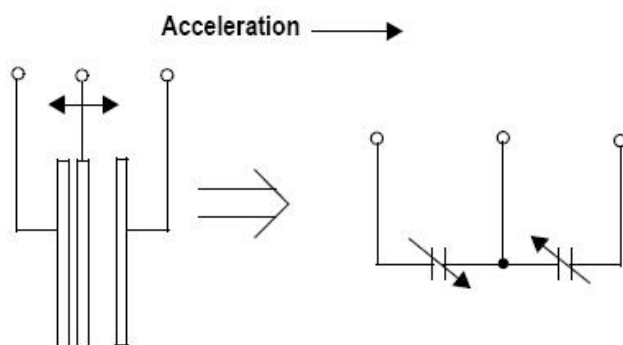
$$C = \varepsilon \cdot \frac{S}{d} [F] \quad (2.3)$$

kde  $C$  je kapacita  
 $\varepsilon$  je permitivita  
 $S$  je plocha elektrod  
 $d$  je vzdálenost mezi elektrodami

Rozsah měření senzorem se spočítá podle rovnice (2.4). Je potřeba vytyčit pouze ta část charakteristiky, která je lineární.

$$F = m \cdot a [N] \quad (2.4)$$

kde  $F$  je síla  
 $m$  je hmotnost tělesa (respektive pohyblivé desky)  
 $a$  je zrychlení



Obrázek 3: Princip akcelerometru [9]

Jednotka naměřené hodnoty senzoru se udává v G/V ( $m/s^2(V)$ ) v případě analogového výstupu a  $m/s^2$  nebo G digitálního výstupu. [16]

### 3.3 Magnetometr

Magnetometr slouží jako čidlo magnetické indukce pole Země. Toto pole se mění v závislosti na času a poloze na Zemi a také na umělých elektromagnetických vlnách vytvářené rádiovým a televizním signálem, rušení síťového vedení, wifi apod. První magnetometr vyrobil německý fyzik a matematik Carl Friedrich Gauss.

Magnetometry se dělí na vektorové a skalární. Skalární měří pouze hodnotu indukce a vektorový pak tuto hodnotu přiřazuje směru, odkud pole vychází. Jako i ostatní čidla má magnetometr mnoho technologických postupů, jak magnetickou indukci měřit. Jedním ze základních je protonový magnetometr, který využívá nukleární magnetické rezonance v atomech vodíku. [10][11]



### **3.4 Barometr**

Barometr je přístroj sloužící k měření atmosférickému tlaku (tlaku ovzduší). Tento tlak je proměnný v závislosti na nadmořské výšce, aktuálního počasí, ale taky se dá uměle nastavit pomocí tlakových komor.

Barometry se dělí na elektrické, kapalinové a mechanické. Používají se především na předpověď počasí, ale také v letectví na stanovení letové výšky. [12]

### **3.5 GPS modul**

Jde o globální družicový systém, kterým se dá přesně určit poloha přijímače, pomocí měření doby šíření signálu od družice k GPS modulu. Je vždy potřeba, aby přijímač byl připojen k více družicím a platí, že čím více je připojených družic, tím lepší je přesnost polohy. Kromě polohy jsou uschovány v signálu další užitečné informace jako je např. aktuální čas, datum atd.[18]

### **3.6 Ostatní**

Další senzory pro zjištění polohy v prostoru jsou jen vypsány s drobným popisem, protože nejsou v této práci použity. Většina z níže popsaných čidel není vhodná pro orientaci v otevřeném prostranství, protože pracují na principu odrazu, který není možné vytvořit na velké vzdálenosti a musí být v cestě předmět pro onen odraz.

#### **3.6.1 Ultrazvukový senzor**

Tento senzor funguje na základě akustických vln. Skládá se ze dvou částí a sice z vysílače a přijímače. Vysílač vyšle akustickou vlnu a definované délce a tvaru a přijímač zachycuje tento signál. Porovnání obou signálů a časové zpoždění mezi nimi určí, jak je předmět, od kterého se zvuk odrazil, daleko.

#### **3.6.2 Infračervené senzory**

Podobně jako ultrazvuk se pomocí infračervené diody vysílá světelný signál, který je přijímán fotodiodou a následným porovnáním obou signálů se zjistí vzdálenost od předmětu. Tyto čidla je vhodné používat jen při šeru či tmě, protože většina světelných zdrojů (žárovky, slunce atd.) obsahuje infračervený paprsek, který vytváří chyby v měření.

#### **3.6.3 Optovláknové senzory**

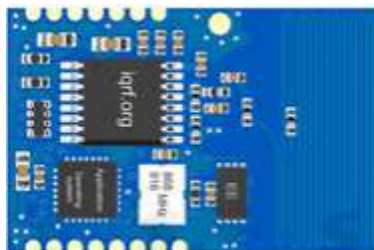
Pracují s difuzním odrazem. Vysílač, v podobě LED diody nebo infraLED vysílá signál, který se následně odrazí od předmětu s matným povrchem do všech stran. Malá část tohoto světelného signálu dorazí ke snímači, který tento signál zachytí a opět se porovnají oba signály.[17]

## 4 Návrh a realizace konstrukce měřicí části s bezdrátovým přenosem

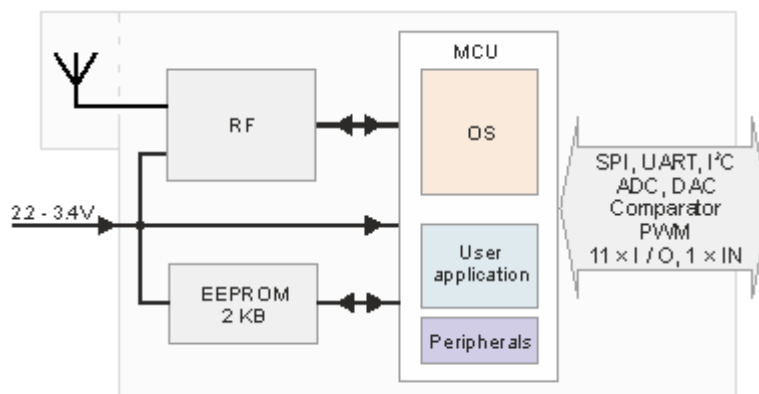
Měřicí část je realizovaná pomocí komunikačního modulu TR-56D, modulu s čidly AltIMU10 a nabíjecím obvodem MCP73831. Zapojení a popis těchto modulů je popsáno níže. Protože se data z měření zpracovávají až v PC, tak nebyla potřeba vysokého výpočetního výkonu procesoru měřicí části.

### 4.1 Komunikační modul IQRF

Pro bezdrátovou komunikaci byly zvoleny komunikační moduly TR-56D od společnosti IQRF, které mají dosah okolo 700m. Moduly se skládají z jednoduchých komponent. Mikroprocesoru a RF obvodu. Je mnoho typů těchto modulů, ale TR-56D, který je v této práci použit se skládá z mikroprocesoru PIC16LF1938 od společnosti Microchip a komunikačního rádiového modulu MRF49XA taktéž od společnosti Microchip. Na obrázku 4 je tento modul zobrazen. Pro větší uživatelský rozlet je připojena k modulu externí EEPROM paměť 2KB. Blokové schéma je na obrázku 5. [13]



Obrázek 4: Modul TR-56D[13]



Obrázek 5: Blokové schéma TR-56D modulu[13]

#### Základní vlastnosti modulu:

- Snadné programování díky OS
- Možnost volby RF komunikace (peer-to-peer nebo IQMESH)

- Dosah až 700 metrů
- Volba komunikační kanál (7 kanálů na frekvenci 868 / 916 MHz)
- Volba přenosová rychlosti (1.2 kb/s, 19.2 kb/s, 57.6 kb/s, 86.2 kb/s)
- Napájení 3V-3,4V
- Spotřeba max. 24mA při nastaveném nejvyšším výkonu RF
- Možnost propojení konektorem SIM nebo napájením do DPS
- Malé rozměry 15.2 mm x 14.9 mm x 2.0 mm
- 1xI/O, 1xVstup, 4xA/D, 1xD/A, 1xPWM
- Komunikace SPI, UART, I<sup>2</sup>C, RF

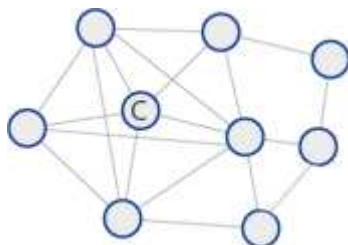
Dosah modulů závisí na mnoha faktorech. Jedním základním faktorem viditelnosti jednotlivých modulů na sebe. Pokud je v cestě překážka, snižuje se rádiový dosah. Dalším důležitým faktorem je rychlost komunikace. Platí, že čím větší je rychlost, tím menší na menší vzdálenost lze komunikovat. S tím souvisí také komunikační kanály, na kterých se komunikuje. Pokud je na daném kanálu mnoho zařízení, dochází ke kolizím.

Jak již bylo řečeno jedná se o IQRF komunikaci. Není potřeba pod tímto názvem hledat nic složitějšího, protože se jedná o RF komunikaci s tím, že je možné využít jednu z lepších variant jako je např. MESH síť, která se jeví jako inteligentní. Odtud pak vznikl název IQRF. Pojďme si ukázat možnosti propojení komunikačních modulů.

**Peer-to-peer** – Nejedná se o síťovou komunikaci, protože mezi sebou komunikují dva nebo více modulů, kde jeden vysílá a ostatní poslouchají. Vysílací modul vyšle data do prostoru a nadále se o příjmu dat jiným zařízením nestará. Operační systém IQRF modulu, který přijímá, tuto zprávu zachytí, ale je nutné, aby programátor tyto data dále zpracoval. Moduly nemají adresy ani žádné další autorizační značky, a proto mají nastavený příjem všechny moduly komunikující na stejném kanálu. Toto řešení je nejjednodušší, ale je nutné v programu ošetřit, která data jsou pro tento modul a která ne.

**IQMESH** – Zde se již jedná o síťovou komunikaci, ve které sice všechny moduly zprávu přijímají, ale OS sám rozhodne, jestli je zpráva pro něj. Následně zprávu přeposílá adresátovi nebo tuto zprávu zahodí. To záleží na nastavení OS. Pokud si zprávy mezi sebou moduly přeposílají sami, jedná se o MESH síť a má jednu velice výbornou vlastnost, a sice, že pokud na sebe fyzicky vysílač a přijímač nevidí, přepoše se zpráva pomocí třetího modulu. Na obrázku 6 je pak tato architektura vidět. C je označený koordinátor, tedy hlavní MASTER, který celou síť řídí, ale nemusí se jednat pouze o vysílač. Je to pouze modul s většími právy, než ostatní moduly nastavené na tzv. NODE. Samotnou síť tedy tvoří koordinátor, který tuto síť řídí a k němu podřízené

node, které mohou komunikovat jen v této síti. Pro komunikaci s jinou sítí soužij koordinátor, který takto může komunikovat pokud je tak nastaven.



**Obrázek 6: MESH síť [9]**

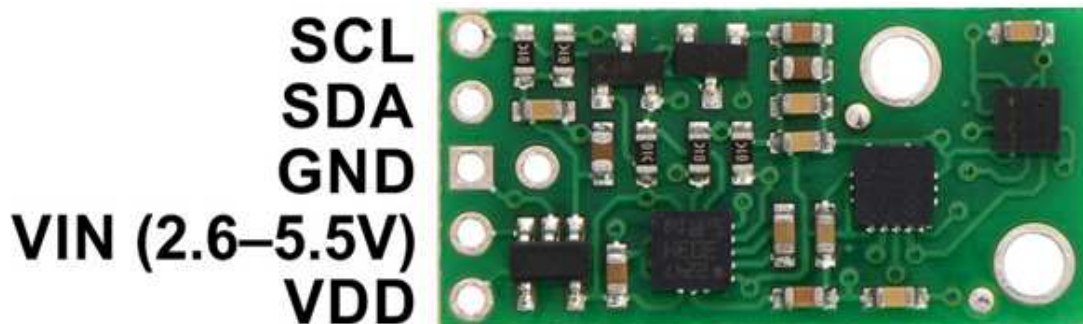
Každý Node má svou adresu a proto je tedy jednoduché zjistit pro koho je daná zpráva určená. Koordinátor může mít pod sebou neinicializovaných 239 Nodů, které vzájemně sváže pomocí tzv. Bonding. Tato funkce vyhledá všechny Node v dosahu a připojí je do dané sítě. Dále je možné se s modulem bavit i když na sebe fyzicky nevidí.

**Použití:**

- Bezdrátové tlačítko/vypínač
- Dálkové ovládání zařízení
- Bezdrátové ovladače (klávesnice, myš, atd.)
- Bezpečnostní technika
- Bezdrátové senzory
- Převodník mezi komunikacemi SPI/RF apod.

## **4.2 Modul s čidly AltIMU-10 v4**

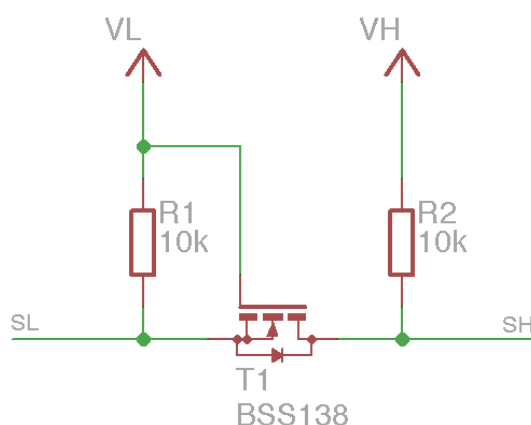
Pro získávání dat pro výpočty orientačních matic byl použit modul AltIMU, který disponuje barometrem (LPS25H), tříosým gyroskopem (L3GD20H), tříosým akcelerometrem, tříosým magnetometrem (LSM303D). Pro větší uživatelský rozlet je zapojen do modulu stabilizátor napětí na 3,3V, který ovšem není nutné využít. Obrázek modulu a jeho komunikační piny jsou na obrázku 7. Je zde také vidět, že orientace všech čidel je stejná, aby se v programu nemusely hodnoty přepočítávat na správnou orientaci. Každé z čidel má v sobě zakomponováno i čidlo teploty, které je buď aktivní nebo vypnuté. Stejně tak je možné některé čidla vypnout a tím snížit spotřebu modulu. [14]



Obrázek 7: Modul AltIMU10[14]

V horní části obrázku 7 je vidět zapojení dvou tranzistorů, která slouží k obousměrné konverze napětí. Přesnější zapojení je na obrázku 8 (respektive je toto zapojení použito dvakrát), ale princip je velice jednoduchý.  $V_L$  se připojuje na nižší napájecí napětí a  $V_H$  na vyšší např.  $V_L$  bude 3,3V a  $V_H$  5V. Při klidovém stavu je sběrnice na logické jedničce tedy pro náš příklad 3,3V a 5V. O tyto úrovně se postarají pull-up rezistory. Rozdíl napětí mezi vývody *source* a *gate* tranzistoru je blízké nule a tranzistor je zavřen.

Jakmile nižší úroveň spojí linku se zemí, rozdíl napětí mezi *source* a *gate* tranzistoru stoupne a tím se tranzistor otevře a logická nula se objeví i na pravé straně. Obdobně je to i v opačném případě. Pokud vyšší úroveň spojí linku se zemí, tak vnitřní dioda zapojena v tomto MOSFETu vytvoří stejný rozdíl napětí mezi *source* a *gate* a tranzistor se opět otevře. Tím se objeví logická nula na nižší úrovni.



Obrázek 8: Zapojení obousměrné konverze napětí

#### Vlastnosti a funkce modulu:

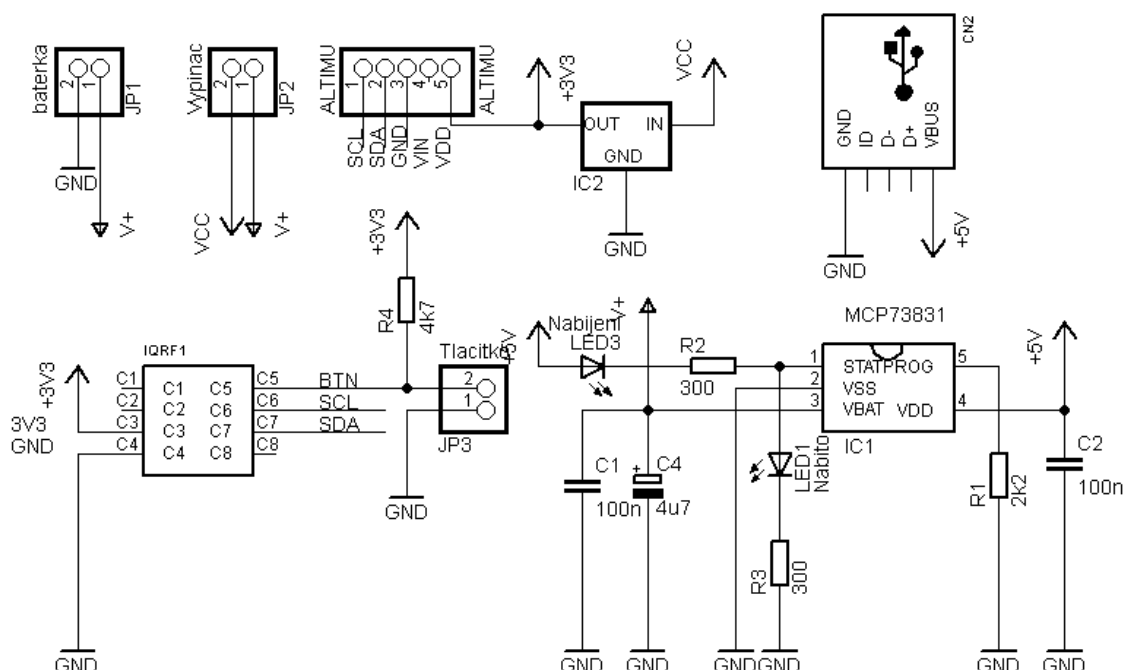
- Malé rozměry 25x13mm
- Napájecí napětí 2,5V-5,5V
- Proudový odběr max. 6mA\*

- I<sup>2</sup>C komunikace
- Dvě různé adresy pro čidla pomocí pinu SA0
- Gyroskop:
  - Tříosý
  - 16b výstup na každou osu
  - Citlivost  $\pm 245$ ,  $\pm 500$  a  $\pm 2000^\circ/\text{s}$
- Akcelerometr:
  - Tříosý
  - 16b výstup na každou osu
  - Citlivost  $\pm 2$ ,  $\pm 4$ ,  $\pm 6$ ,  $\pm 8$  a  $\pm 16$  g
- Magnetometr:
  - Tříosý
  - 16b výstup na každou osu
  - Citlivost  $\pm 2$ ,  $\pm 4$ ,  $\pm 8$  a  $\pm 12$  gauss
- Barometr:
  - 24b výstup
  - Citlivost od 260 mbar do 1260 mbar (26 kPa to 126 kPa)

\*podle nastavení sleep modů jednotlivých čidel nebo jejich vypnutí

### 4.3 Schéma propojení modulů

Schéma pro propojení obou modulů je jednoduché, protože většina součástek potřebných pro funkci jednotlivých modulů je na samotných modulech. Bližší popis těchto modulů byl uveden výše v kapitolách 3.1 a 3.2. Schéma zapojení je uvedeno na obrázku 9. Do obvodu byl přidán nabíjecí obvod s externími součástkami a stabilizátor napětí na 3,3V. Nabíjecí obvod má zapojené dvě kontrolní LED diody, které signalizují stav nabití. Červená LED2 svítí při nabíjení a Zelená LED1 při stavu nabito. Nabíjení je možné pouze po připojení micro USB. Ostatní součástky na DPS jsou pouze piny pro připojení baterky, vypínače a tlačítka. Kvůli napěťovému konvertoru jež je popsán v kapitole 3.2 nebylo možné použít vnitřní stabilizátor modulu AltIMU pro napájení modulu TR-56D, a proto musel být přidán externí stabilizátor. Problémem bylo vstupní napětí baterie, které se pohybuje v rozmezích 3,5 – 4,2V, které není možné připojit na vstupní piny modulu TR-56D (procesoru pic16). Naskýtalo se jednoduché řešení v podobě přepájení pull-up rezistorů na straně většího napětí k menšímu, ale byl by to zásah do nového modulu. Proto jsme se rozhodli použít externí stabilizátor. Funkce tlačítka bude popsána níže.



Obrázek 9: Schéma propojení modulů

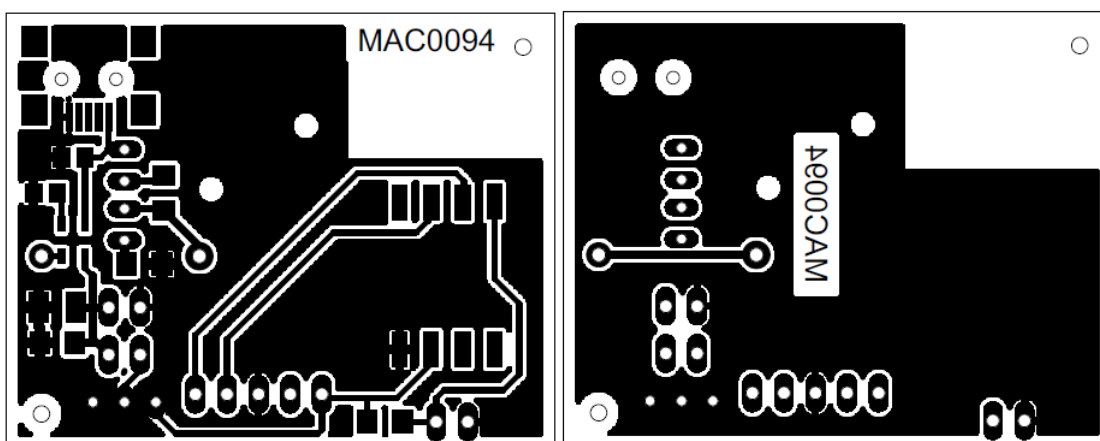
#### 4.4 DPS a krabička

Především se dbalo na minimalizaci toho zařízení a to vedlo k výběru mají krabičky i rozmístění součástek po DPS.

##### 4.4.1 DPS

DPS byla vytvořena podle vnitřních rozměrů krabičky, které jsou 46x35mm. Tyto rozměry jsou dostačující pro tak malé zařízení. Oboustranná DPS je zobrazena na obrázku 10. Jedná se o oboustrannou desku, přičemž strana BOTTOM je pouze pro lepší propojenost zemnicí cesty a spojení jedné propojky. Pro testovací část nebylo nutné vyrábět oboustranný plošný spoj, nicméně pro sériovou výrobu by byla tato volba velkou výhodou. Větší plocha a kratší propojení zemnicí cesty má příznivý výsledek na okolní rušení i na rušení vytvářené samotným zařízením.

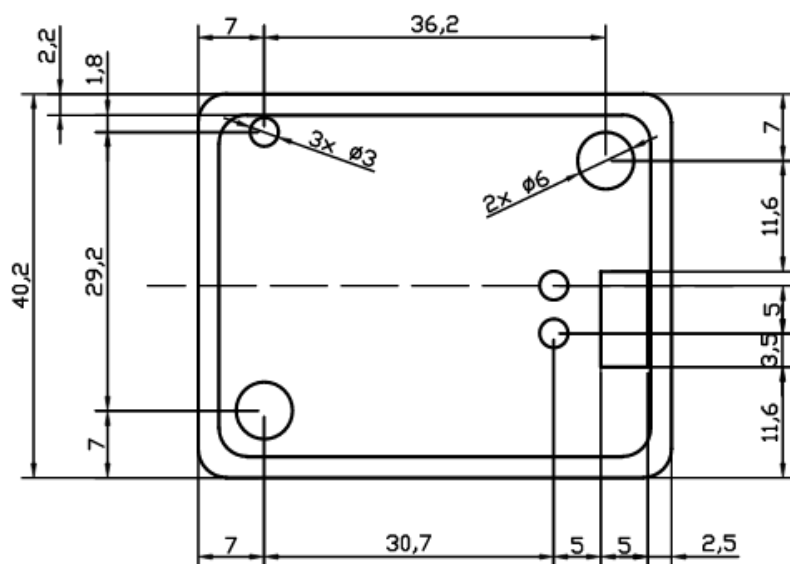
Součástky byly použity především SMD, které bývají menších rozměrů a to vedlo k minimalizaci DPS. Klasické součástky jsou pouze konektory pro baterii, vypínač, tlačítko a modul AltIMU. Posledním zástupcem klasických součástek osazených na této DPS jsou signalizační LED diody a stabilizátor na 3,3V. V pravém rohu DPS je vyleptaná měď kvůli lepšímu dosahu antény. Kdyby tomu tak nebylo, tak větší část vysílacího výkonu se v zemnicí cestě ztratí. Osazovací plánec a ostatní potřebné plánky pro výrobu DPS se nacházejí v příloze 1.



Obrázek 10: DPS vlevo TOP, vpravo BOTTOM

#### 4.4.2 Krabička

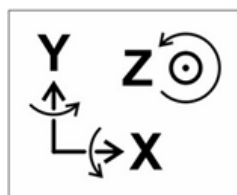
Krabička byla zvolena menších rozměrů pro minimalizaci měřicí stanice a to 40x50x20mm. Výkres krabičky je pak v příloze 3. Krabička se skládá ze dvou částí a to „vaničky“ a víka. Do „vaničky“ byla vložena baterie a osazená DPS. Mezi ně byla potřeba vložit kousek izolačního papíru, aby nedocházelo ke zkratům. DPS je ke krabičce přišroubována samopřeznými šrouby do připravených otvorů podle výkresu. Do Krabičky bylo nutné vyvrtat otvory pro USB konektor a do víka vyvrtat díry na obě LED diody, tlačítko a vypínač. Výkres opracovaného víka je na obrázku 11. Vypínač je k víku přilepen tavnou pistolí, aby pevně držel s víkem.



Obrázek 11: Výkres víka

Na obrázku 12 je vidět štítek s orientací os, který byl přilepen na měřicí modul.

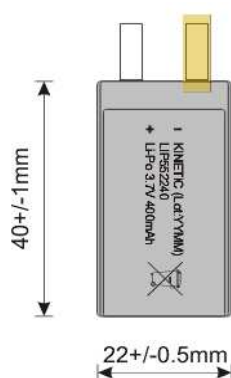




Obrázek 12: Štítek orientace os

#### 4.5 Akumulátor a nabíječka

Jako akumulátor byl zvolen typ LIP552240. Jedná se o Li-Po (lithium-polymerový) akumulátor s napětím 3,7V/400mAh a malými rozměry (22x40x5,5mm). Baterie je na obrázku 13.



Obrázek 13: Použitý akumulátor[19]

Tento akumulátor je v této práci možné nabít pouze pomocí micro USB konektoru, který je připojen k nabíjecího obvodu MCP73831. Schéma zapojení tohoto obvodu je na obrázku 9 a jde o standardní zapojení obvodu. MCP73831 má nabíjecí napětí 4,2V, což je přesně podle parametrů baterie. K obvodu je připojeno několik externích součástek, které lze vyčíst z datasheetu tohoto obvodu, ale tím se nebudeme zabývat. Pro nastavení nabíjecího proudu je připojen odpor R1, kterým nastavuje nabíjecí proud podle vztahu (3.1). [19]

$$I_{REG} = \frac{1000V}{R1} = \frac{1000}{2200} = 0,454545A = 454mA \quad (3.1)$$

kde  $I_{REG}$  je nabíjecí proud

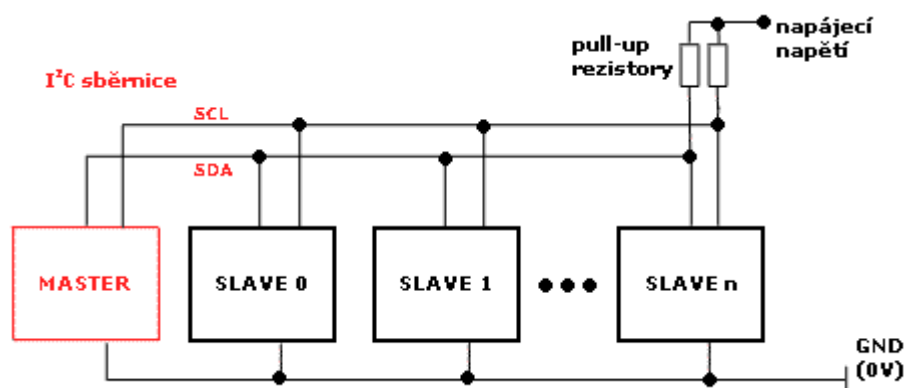
$R1$  hodnota odporu R1

LED1 a LED2 slouží jako signalizační LED, kde LED2 svítí při nabíjení a LED1 při stavu nabito. Kondensátory pak slouží jako filtrace vstupního a výstupního napětí.

## 4.6 Komunikace mezi moduly

Komunikace mezi moduly byla zvolena I<sup>2</sup>C, protože měřicí modul AltIMU má na výběr pouze tuto volbu. Samotné čidla umí komunikovat i po SPI, ale modul je propojen na nejjednodušší variantě, na které SPI není dovolena. Měřicí modul není tak univerzální, ale je minimalizován na minimum součástek i na celkový rozměr.

Jak již bylo řečeno, moduly mezi sebou komunikují přes I<sup>2</sup>C neboli IIC a nesprávně označovanou I2C. Tento komunikační protokol je jednoduchý, ale obsahuje řídící kroky pro režii. Základem je nastavení řídicího MASTER, který řídí komunikaci. Tím je mikroprocesor PIC16 zapojený na komunikačním modulu TR-56D. Ostatní čidla jako gyroskop, akcelerometr/magnetometr a barometr jsou nastaveny jako SLAVE jak je vidět na obrázku 14. [15]



Obrázek 14: Blokové zapojení I<sup>2</sup>C komunikace[15]

Na sběrnici je možné připojit až 128 zařízení, které mají svou adresu, a proto se MASTER baví vždy jen s jedním SLAVE. To v případě více mikroprocesorů zapojených jako SLAVE umožňuje nesledovat sběrnici, pokud nebyli osloveni adresou právě oni a mohou tak vykonávat jinou činnost. Adresa může být 7b nebo 10b. Při použití 7b adresy můžeme zapojit na sběrnici až 128 zařízení, což pro většinu aplikací s nadhledem stačí, ale je možnost komunikovat s 10b adresou, což umožňuje použít až 1024 zařízení. Rychlost sběrnice je standardně 100kHz, ale u většiny čipů je možnost nastavit vyšší rychlost 400kHz nebo dokonce 1MHz, ale je potřeba si tyto rychlosti ověřit u všech zapojených zařízení.

Tedy k samotné komunikaci mezi zařízeními. MASTER generuje hodiny na vodič SCL při každém pokusu o komunikaci. Existuje několik komunikačních stavů/prvků.

- **Stav klidu** - je stav, při němž neprobíhá žádná komunikace. MASTER negeneruje hodiny a oba vodiče jsou v logické jedničce pomocí pull-up rezistorů.

- **Start bit** - Jde o zahájení přenosu nebo opakované zahájení. Je generován tak, že na SDA vodiči se logická úroveň z 1 do 0 při SCL na hodnotě 1.
- **Stop bit** - Ukončuje přenos opačným způsobem než start bit. Tedy mění hodnotu SDA z 0 do 1 při SCL na hodnotě 1.
- **Přenos dat** - Data mají vždy 1Byte (8b od nejvyššího po nejnižší). Při každém zákmitu (z logické 1 do 0 a zpět do logické 1) na SCL je přenesen na SDA právě jeden bit.
- **Adresa** - je standardně 7b. Posledním odesílaným bitem adresy je R/W bit, který určuje, jestli chceme od SLAVE čtení nebo zápis.
- **Ack bit** - Jde o potvrzovací bit správnosti komunikace. Tento bit vždy generuje ta stanice, která přijímala data/adresu (při vysílání adresy Ack odesílá SLAVE). Pochopitelnější je to z obrázku 15.



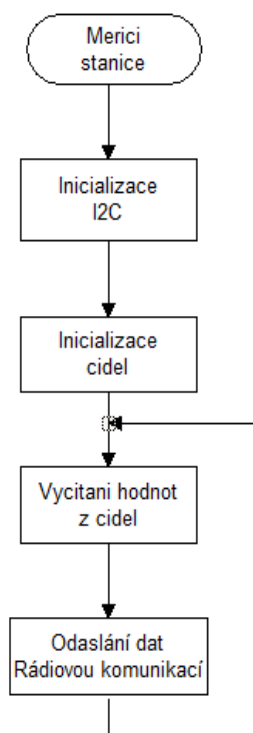
Obrázek 15: Rámec komunikace I<sup>2</sup>C [15]

Jak je vidět na obrázku 15, komunikace probíhá oběma směry. Nejdříve se na sběrnici vyšle start bit a následně adresa zařízení, se kterým chceme komunikovat a bit R/W pro určení, jestli chceme do SLAVE zapsat nebo z něj číst. Následně oslovené zařízení musí odpovědět potvrzovacím bitem Ack. Pokud to SLAVE udělá, začne zápis/čtení, které se vždy potvrdí bitem Ack a vždy tak, že Ack bit odesílá ten, kdo data přijímal. Nakonec se ukončí komunikace pomocí stop bitu a sběrnice je ve stavu klidu.

#### 4.7 Software měřicí částí

Program pro měřicí část byl vytvořen ve vývojovém prostředí IQRF IDE 4.17. Toto prostředí je navrženo pro kompilaci a nahrávání programu do IQRF modulu jako je např. modul TR-56D a spolupracuje s programem Notepad++, ve kterém se píše samotný program v jazyce C.

Kód pro měření a řízení měřicí části se skládá z jednoduchých částí. Tento kód je samostatná aplikace v operačním systému, který běží na pozadí modulu TR-56D. OS se stará o rádiovou, I<sup>2</sup>C komunikaci a mnoho dalšího. Na obrázku 16 je vývojový diagram tohoto programu a jeho bližší popis je uveden níže. První část programu je nastavení a funkce pro I<sup>2</sup>C komunikaci. Druhá část je pak nastavení a čtení z čidel gyroskopu, akcelerometru a magnetometru. Celý program je k nahlédnutí v příloze 4.



**Obrázek 16: Vývojový diagram měřící části**

#### **4.7.1 První část**

Součástí OS nejsou ovladače pro I<sup>2</sup>C komunikaci, proto bylo nutné vytvořit několik funkcí, které tuto komunikaci zprovozní. Jde o funkce *i2c\_init()*, *i2c\_waitForIdle()*, *i2c\_start()*, *i2c\_repStart()*, *i2c\_stop()*, *i2c\_read(uns8 ack)*, *i2c\_write(uns8 i2cWriteData)*. Tyto funkce slouží jako inicializace a komunikační bity podle kapitoly 3.5. Funkce *i2c\_read* a *i2c\_write* je potřeba pro zápis/čtení.

#### **4.7.2 Druhá část**

Zde je inicializace všech čidel a následné čtení z čidel a odesílání hodnot přes RF v nekonečné smyčce. Hodnoty se odesílají každých 20ms, což je dostatečný čas pro zachycení pohybových změn modulu. Rychlejší události, jako je chvění, nebudou pro systém brány v potaz.

#### **4.7.3 Nastavení registrů čidel**

Čidla je možno nastavit na různou citlivost, komunikaci apod., proto je zde uvedeno konečné nastavení použité v této práci.

**Gyroskop** – Naměří data s frekvencí 95Hz a jsou zapnuty všechny tři osy (X,Y,Z). Dále je potřeba nastavit měření pouze když jsou předchozí data přečtena a citlivost. Ta byla zvolena 500°/s. Gyroskop komunikuje po I<sup>2</sup>C komunikaci. Ostatní

nastavovací registry není třeba nastavovat, protože po zapnutí napájení se automaticky nastaví na defaultní hodnoty.

**Akcelerometr/Magnetometr** – Obě čidla jsou ve společném pouzdře a proto mají i společné registry. Taktéž jako gyroskop jsou nastaveny čidla na I<sup>2</sup>C komunikaci a registry, které zde nejsou uvedeny se nastavují automaticky na defaultní hodnoty. Obě čidla mají zapnuté všechny tři osy, citlivost je nastavena na -4g až 4g respektive -4gauss až 4gauss a měří data s frekvencí 50Hz.

**Měření teploty** – není zapnuto ani u jednoho z čidel a barometr je nastaven do sleep mode, aby zbytečně nezvyšoval spotřebu modulu.

#### **4.8 Operační systém modulu TR-56D**

Jak již bylo výše uvedené, tak modul TR-56D má v sobě nainstalovaný operační systém, který řídí běh celé aplikace a obsahuje RF, SPI a komunikaci s externí pamětí. Dále obsahuje funkce pro LED diody možnost RF programování a mnoho dalšího. Aplikace napsaná pro měřicí část je pouze přidána do tohoto OS, který ji spustí a dále řídí celý běh všech dalších funkcí. [20]

## 5 Návrh a realizace obslužného software pro zobrazení a archivaci dat na PC

Vizualizace byla vytvořena v programovacím prostředí LabVIEW od společnosti NATIONAL INSTRUMENTS. Toto programovací prostředí disponuje mnoha bloky, kterými se „píše“ program. Ve skutečnosti se jednotlivé bloky spojují pomocí čar různých barev. Toto spojení dává výslednému programu funkcionalitu, kterou programátor zamýšlí. Barvy čar znázorňují různé datové typy dat, které daným místem protékají.

Pro jakoukoliv pomoc ohledně programování v tomto jazyce naleznete na stránkách NI.com, kde je nejen popis jednotlivých funkčních bloků, ale také příklady aplikací a použití jednotlivých bloků v nich.[21]

### 5.1 Popis programovacího prostředí

LabVIEW se dělí na dvě základní části:

- 1) **Front Panel** - Čelní panel neboli to, co uživatel uvidí při spuštění hotové aplikace
- 2) **Block diagram** - Zde se vytváří funkcionalita programu

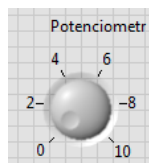
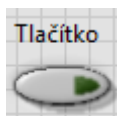
Po napsání programu a odzkoušení jeho funkčnosti lze pomocí LabVIEW vytvořit .exe soubor (spustitelnou aplikaci) nebo z této aplikace vytvořit instalaci dané aplikace. Popis postupu je uveden v kapitole 4.1.3.

#### 5.1.1 Front Panel

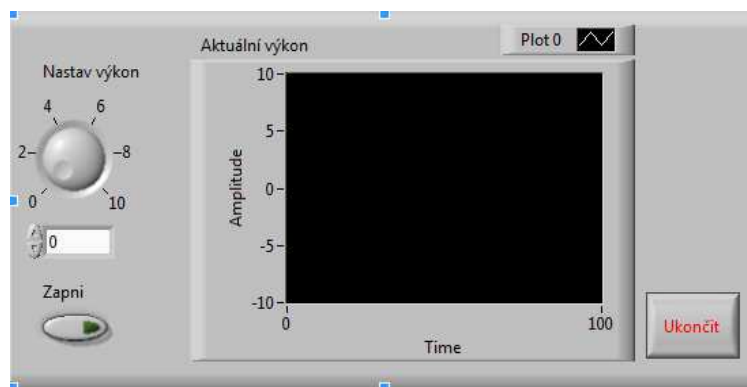
Čelní panel je jediný obraz, který uživatel uvidí při běhu aplikace a není tedy možné se podívat na kód aplikace. Jeho hlavním účelem je uživateli danou aplikaci zjednodušit na jednotlivé prvky, kterými aplikaci ovládá.

Mezi základní prvky patří :

- Tlačítka
- Potenciometr
- Přepínače
- Grafy
- Číselné vstupy



Výsledný panel lze pak ještě graficky upravit pomocí různých dekorací. Ukázka čelního panelu aplikace je na obrázku 17.



Obrázek 17: Ukázka čelního panelu

### 5.1.2 Block diagram

Blokový diagram slouží pro napsání programu, respektive propojení jednotlivých bloků do funkčního celku. Jako v každém programovacím jazyce i v LabVIEW základní proměnné typu:

- int 8,16,32,64
- uint 8,16,32,64
- double, float
- string
- boolean

Dalšími podobnostmi s ostatními jazyky je příkaz switch nebo smyčky while a for, které fungují naprosto stejně jako například v programovacím jazyce C. Rozebírat význam jednotlivých bloků nemá smysl, protože jejich popis je velmi dobře popsán a rozebrán na stránkách NI.com, kde jsou i exampley. Naučit se programovat v LabVIEW není lehká záležitost, ale po získání trochu zručnosti lze vytvořit i složitější aplikace jednoduše.

Hlavní výhodou LabVIEW je možnost použití takzvaných SubVI, tedy rozdělení celého programu do jednotlivých funkčních bloků. Další výhodou je použití .c nebo .m kódu přímo do své aplikace, kde je možné použít již vyvinuté kódy (aplikace) jiných programovacích jazyků a nepřepisovat je do LabVIEW kódu.

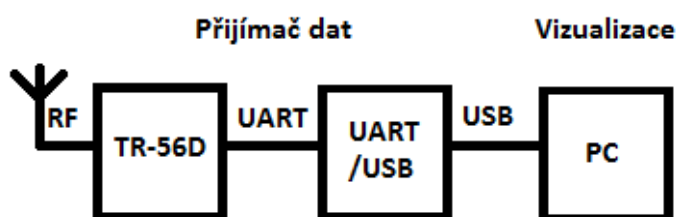
### 5.1.3 Vytvoření .exe souboru a instalačního souboru

Z každého kódu vytvořeném v LabVIEW lze samozřejmě vytvořit aplikace .exe pro jednoduché spouštění v našem operačním systému. Námi vytvořená aplikace lze spouštět přímo v LabVIEW, ale není moc vhodné mít v každém počítači nainstalované toto programovací prostředí. Pro spuštění .exe souboru bez nainstalovaného LabVIEW je nutné doinstalovat *Run-Time Engine*, který umožní spuštění, a proto je vhodnější vytvořit instalaci naší aplikace a tím se tyto problémy vyřeší, protože jsou všechny potřebné soubory součástí instalace.

Vytvoření .exe nebo instalačního souboru je velice jednoduché. Po založení projektu v LabVIEW je dole Build Specification. Pravý klik myši objeví možnosti, které lze s projektem dělat včetně výše zmíněného vytvoření .exe nebo instalačního souboru. Obě volby jsou přednastaveny a budou fungovat, i když nic nezměníme, ale je vhodné si vytvořit vlastní nastavení podle svého gusta. Pro toto nastavení je vhodné prozkoumat stránky NI.com, kde je popis všeho. [21]

## 5.2 Sběr dat

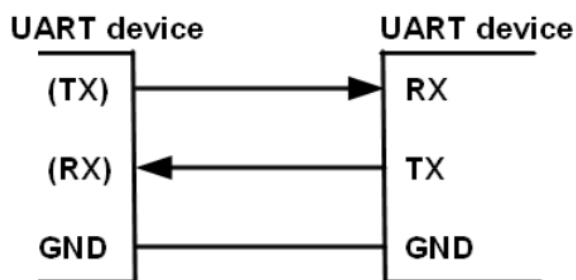
Sběr dat je prováděno spojením modulu TR-56D a převodníku USB/UART podle obrázku 18. Jedná se pouze o mezičlánek pro zpracování dat na PC aplikaci.



Obrázek 18: Propojení modulů

### 5.2.1 Komunikace UART

UART neboli sériová linka je jednou z nejjednodušší komunikací vůbec. Jedná se o propojení dvou zařízení pomocí dvou, respektive tří drátů (Rx, Tx, GND), kterými mohou probíhat signály oběma směry zároveň. Propojení zařízení je křížové, nejlépe kroucené dvojlinky podle obrázku 19. Křížové propojení znamená, že TX zařízení 1 vstupuje do vstupu RX druhého zařízení a naopak.



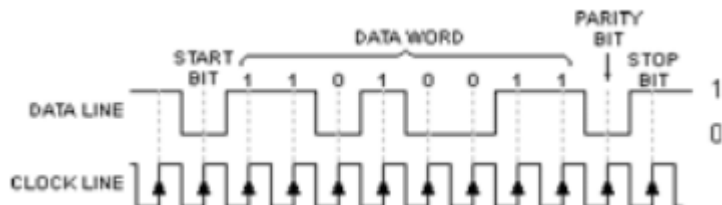
Obrázek 19: Propojení zařízení pomocí UART[23]

Přenos může být synchronní nebo asynchronní. Rozdíl je, jak samotný název napovídá, v synchronizaci jednotlivých dat.

**Synchronní přenos** – Každý přenesený bit musí být potvrzen impulzem, to ale vytváří nutnost stanovit kdo tento synchronizační impuls vyšle. Jednodušší cesta je asynchronní přenos.



**Asynchronní přenos** – Odesílají se celé bajty a mezi těmito bajty jsou dva (nebo víc) znaků, a to Start a Stop bit. Do komunikace se ještě často přidává paritní bit pro určení, jestli došly data správně. Komunikační rámec je na obrázku 20.



Obrázek 20: Komunikační rámec [23]

Funkcionalita režijních bitů:

- **Start bit** – Změna logické úrovně z logické 1 do 0
- **Stop bit** – Změna logické úrovně z logické 0 do 1
- **Parita** – Parita je kontrolní součet všech bitů a nastavuje se podle přednastavení na sudou, lichou nebo žádnou. Jedná v podstatě o součet všech bitů, které nesou informaci v logické 1 a následně se nastaví parita. Např. Odeslaný bajt se skládá ze samých nul (00000000), tak se parita nastavená na sudou nastaví také do 0, protože obsah logických 1 je sudý počet. [23]

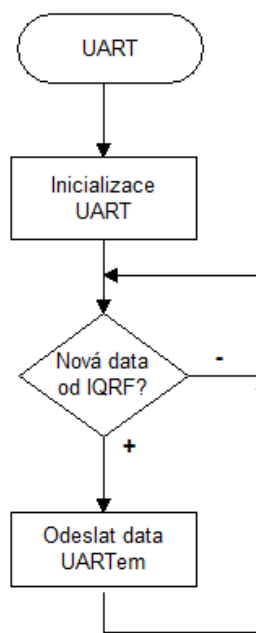
### 5.2.2 Nastavení UARTu

Protože měřicí moduly odesílají data každých 20ms, je potřeba posílat data tolikrát rychleji, kolik měřících modulů připojíme. Pro odesílání dat každých 20ms odpovídá rychlosti 10500Hz. Pro možnost připojení více měřících stanic byla rychlost UARTu modulu TR-56D nastavena na 38400 baudů, což odpovídá připojení maximálně 3 stanic. Výpočet je uveden na rovnici (4.1).

$$f_{\max} = b \cdot B \cdot \frac{1}{t_k} = 10 \cdot 21 \cdot \frac{1}{0,02} = 10500 \text{ Hz} \quad (4.1)$$

- kde  $f_{\max}$  je maximální frekvence přenesených dat  
 $t_k$  je čas mezi odeslanými vzorky  
 $b$  je počet přenesených bitů je 8 + start a stop bit(parita není použita)  
 $B$  je počet přenesených bajtů(byte)

Podobně jako u I<sup>2</sup>C komunikace bylo nutno implementovat funkce podporující komunikaci UART, protože OS ji nepodporuje. Tyto funkce byly přepracovány z původního example UART\_LINK.c dostupný ze stránek iqrf.org. Celý kód je uveden v příloze 5. Vývojový diagram je pak na obrázku 21.



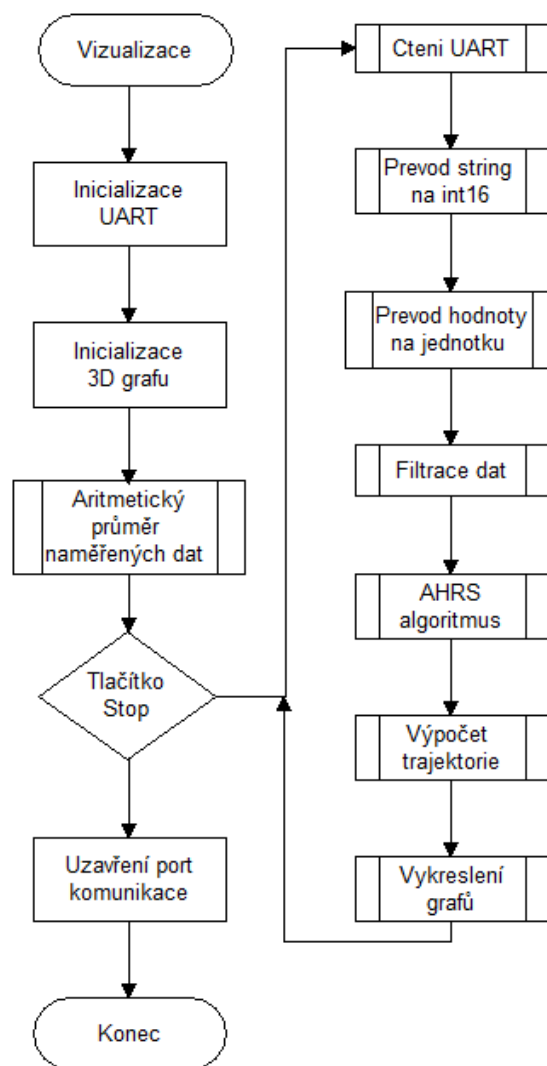
Obrázek 21: Vývojový diagram

### 5.3 Vizualizace

Vytvořená vizualizace se dá rozdělit do několika částí. Tyto části jsou popsány v kapitolách níže. Jak je popsáno výše, tak vizualizace je vytvořena v programovacím prostředí LabVIEW.

#### 5.3.1 Program vizualizace

Celý program je v příloze 6, nicméně na obrázku 22 je vývojový diagram celého programu. Inicializace se skládá z nastavení a otevření příslušného portu a nastavení počátečních podmínek pro 3D graf. Pro přesnější výpočty byla použita funkce *Init Offset*, která zprůměrní 100 hodnot z jednotlivých čidel a s těmito hodnotami se nadále pracuje jako s konstantním offsetem. Čidla mají i v klidovém stavu offsety, které jsou chybové členy a je nutné je pro správný výpočet odečíst od aktuální hodnoty.



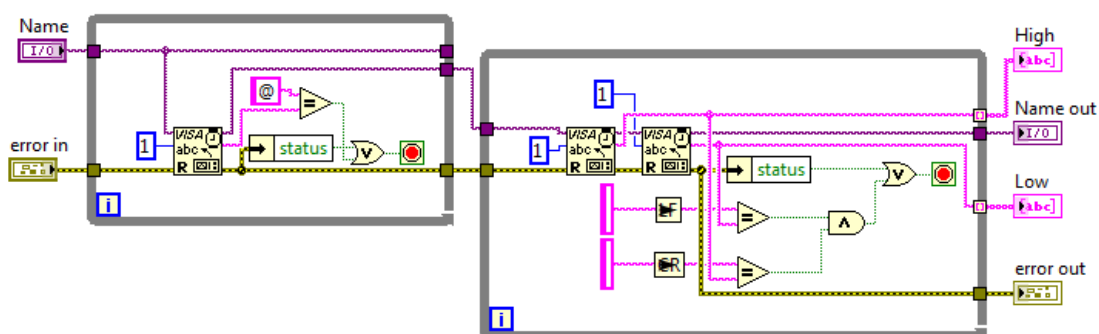
Obrázek 22: Vývojový diagram vizualizace

Hlavní část programu je v cyklicky volána každých 20ms, což je i interval odesílaných dat z vysílače. Data se přečtou a následně převedou pomocí funkcí *Ctení Uart* a *STR to INT16*. Dále je potřeba hodnoty čidel převést na jednotku pomocí přepočtových konstant ve funkci *Prevod Cidel*. Následně se data odfiltrují pomocí jednoduchého Kalmanova filtru a převedou se hodnoty z čidel do rotace(AHRS) respektive posunu v osách(*Posun Osy*). Jednotlivé funkce jsou popsány níže.

Výstupem tohoto programu jsou grafy v závislostech hodnota (akcelerometru, gyroskopu a magnetometru) na čase, 3D rotace objektu a 3D trajektorie objektu. Hodnotou je myšlena hodnota daného čidla např. zrychlení všech tří os uvedené v jednotkách G, kde  $1G = 9,81m/s^2$ .

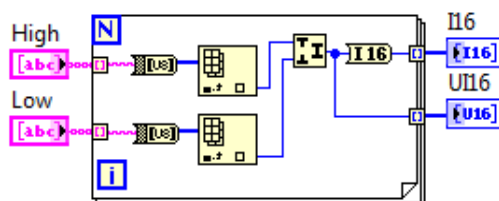
### 5.3.2 Popis jednotlivých funkcí

**Ctení Uart** je funkce, která má za úkol přečíst data ze sériové linky a rozdělit je na dvě pole. Prvním polem jsou hodnoty High bajtu (horní polovina 16b čísla) a druhé pole jsou hodnoty Low bajtu (dolní polovina 16b čísla). Protože UART neumí komunikovat s více bajty najednou bylo nutné toto 16b číslo rozdělit na dvě samostatné hodnoty. Program funkce je na obrázku 23, kde jsou vidět dva cykly while. První cyklus hledá v přijatých datech @, který slouží jako adresa měřícího modulu a také tento znak znázorňuje začátek celého řetězce. Druhá smyčka while ukládá hodnoty do pole a poslední znaky, které přijdou musejí odpovídat ASCII znakům CR a LF (návrat kurzoru na začátek a odřádkování). Po vykonání těchto dvou cyklů je tato funkce ukončena.



Obrázek 23: Funkce Ctení Uart

**STR TO INT16** tato funkce sloučí a převede obě pole na 16b číslo datového typu int16. Protože UART přenáší bajt v datovém typu string, musí tato funkce ještě převést string na int16 číslo. Zde je použita funkce for, která převádí a spojí High a Low čísla a uloží je opět do pole. Při nevyplnění N (počtu provedení tohoto cyklu) funkce for poběží tak dlouho, dokud bude mít vstupní pole nějakou hodnotu. Pole končí hodnotou NULL, na kterou tato funkce nereaguje.

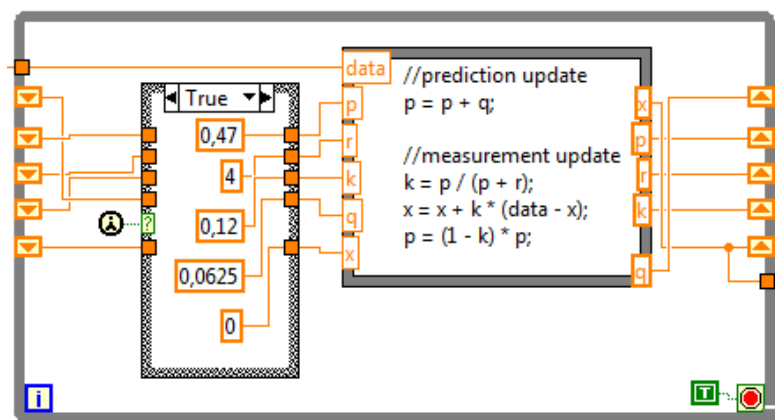


Obrázek 24: Funkce STR TO INT16

**Prevod cidel** je funkce, která upravuje přijaté číslo na jednotku daného čidla. Jde o vynásobení hodnoty konstantou, která je stanovena podle nastavené citlivosti daného čidla.

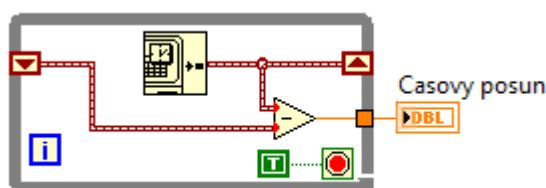
**Filtrace dat** má za úkol vyfiltrovat data pomocí Kalmanova filtru. Zde je použit cyklus while, který se ale vykoná vždy jen jednou po zavolání. Je to pouze pro uložení předchozí hodnoty. Uvnitř tohoto cyklu je blok, který nastavuje počáteční podmínky

konstant (vlevo na obrázku) a blok, který vykonává kód napsaný v jazyce C (vpravo na obrázku). Popis jak funguje Kalmanův filtr je v následující kapitole.



Obrázek 25: Kalmanův filtr

**Casovy posun** je funkce pro vytvoření časového rozdílu mezi přijatými daty. Data jsou odesílána každých 20ms, ale některá data nedorazí, a proto je potřeba zjišťovat skutečný čas mezi vzorky.



Obrázek 26: Funkce časový posun

**AHRS** je nejdůležitější funkcí na celé práci. Jedná se o výpočet kvaternionů (čtveřice), ze kterých se pak vypočítává rotace jednotlivých os (Pitch, Roll, Yaw). Kvaternion je rozšíření oboru komplexní roviny. Zatímco komplexní rovina má prvek  $i$  a jedinou podmínku  $i^2 = -1$ , tak kvaternion má prvky  $i, j, k$  a následující podmínky:

$$\begin{aligned} i^2 &= j^2 = k^2 = ijk = -1 \\ ij &= k, & ji &= -k, \\ jk &= i, & kj &= -i, \\ ki &= j, & ik &= -j. \end{aligned}$$

Výpočet kvaternionů vede k jednoduššímu výpočtu rotace os než-li u výpočtu pomocí Eulerových úhlů. Je to dáno tím, že v dynamickém systému vede výpočet přes Eulerovy úhly na diferenciální rovnice, kdežto u výpočtu kvaternionů pouze na lineární rovnice. Také skládání rotace odpovídá jen násobením příslušných kvaternionů.

Nicméně jak vypadá výpočet pomocí dat z čidel? Nejdříve je potřeba zjistit, jestli došly data od akcelerometru a magnetometru v pořádku (musí být nenulová). Pokud jsou špatná, tak se jejich příspěvek do výpočet nezapočítá a vypočítávají se

kvaterniony pouze pomocí dat z gyroskopu. Natočení gyroskopu odpovídá natočení měřicího modulu, ale pro přesnější výpočet je potřeba dopočítat rotaci i pomocí ostatních čidel. Tyto hodnoty se nejdříve nejdříve normalizují pomocí vzorce (4.1), kde se tento vzorec aplikuje na každou osu daného čidla, kromě gyroskopu.

$$ax = ax \cdot \sqrt{ax^2 + ay^2 + az^2} \quad (4.1)$$

kde  $ax$  hodnota akcelerometru osy X

$ay$  hodnota akcelerometru osy Y

$az$  hodnota akcelerometru osy Z

Dále se vypočítají pomocné hodnoty  $halfvx$ ,  $halfvy$ ,  $halfvz$  sloužící k úpravě výpočtu gravitačního pole země a  $halfwx$ ,  $halfwy$ ,  $halfwz$  a  $halfex$ ,  $halfey$ ,  $halfez$  pro úpravu výpočtu magnetického pole země, podle níže uvedených vztahů. Proměnná  $q0$  až  $q3$  jsou výše zmíněné kvaterniony, které jsou čteny z předchozího stavu a následně je dopočítán jejich nynější stav.

#### Předpokládaný směr gravitačního pole:

$$\begin{aligned} halfvx &= q1 \cdot q3 - q0 \cdot q2 \\ halfvy &= q0 \cdot q1 + q2 \cdot q3 \\ halfvz &= q0 \cdot q0 - 0,5 \cdot q3 \cdot q3 \end{aligned} \quad (4.2)$$

#### Reference směru magnetického pole Země:

$$\begin{aligned} hx &= 2(mx(0,5 - q2 \cdot q2 - q3 \cdot q3) + my(q1 \cdot q2 - q0 \cdot q3) + mz(q1 \cdot q3 + q0 \cdot q2)) \\ hy &= 2(mx(q1 \cdot q2 + q0 \cdot q3) + my(0,5 - q1 \cdot q1 - q3 \cdot q3) + mz(q2 \cdot q3 - q0 \cdot q1)) \\ bx &= \sqrt{hx^2 + hy^2} \\ bz &= 2(mx(q1 \cdot q3 - q0 \cdot q2) + my(q2 \cdot q3 - q0 \cdot q1) + mz(0,5 - q1 \cdot q1 - q2 \cdot q2)) \end{aligned} \quad (4.3)$$

#### Předpokládaný směr magnetického pole:

$$\begin{aligned} halfwx &= bx(0,5 - q2 \cdot q2 - q3 \cdot q3) + bz(q1 \cdot q3 - q0 \cdot q2) \\ halfwy &= bx(q1 \cdot q2 - q0 \cdot q3) + bz(q0 \cdot q1 + q2 \cdot q3) \\ halfwz &= bx(q0 \cdot q2 + q1 \cdot q3) + bz(0,5 - q1 \cdot q1 - q2 \cdot q2) \end{aligned} \quad (4.4)$$

**Chyba je součtem odhadovaného směru a naměřeného směru obou polí Země:**

$$\begin{aligned} halfex &= (ay \cdot halfvz - az \cdot halfvy) + (my \cdot halfwz - mz \cdot halfwy) \\ halfey &= (az \cdot halfvx - ax \cdot halfvz) + (mz \cdot halfwx - mx \cdot halfwz) \\ halfez &= (ax \cdot halfvy - ay \cdot halfvx) + (mx \cdot halfwy - my \cdot halfwx) \end{aligned} \quad (4.5)$$

**Aplikace chyby na data z gyroskopu:**

$$gx = gx + halfex$$

$$gy = gy + halfey$$

$$gz = gz + halfez$$

(4.6)

**Úprava hodnot gyroskopu podle vzorkovací frekvence:**

$$gx = gx \cdot 0,5 / SampleFreq$$

$$gy = gy \cdot 0,5 / SampleFreq$$

$$gz = gz \cdot 0,5 / SampleFreq$$

(4.7)

**Uložení mezivýsledku q0, q1 a q2 pro následující zpracování:**

$$qa = q0$$

$$qb = q1$$

$$qc = q2$$

(4.8)

**Výpočet kvaternionů z upravených dat gyroskopu:**

$$q0 = q0 + (-qb \cdot gx - qc \cdot gy - q3 \cdot qz)$$

$$q1 = q1 + (qa \cdot gx + qc \cdot gz - q3 \cdot gy)$$

$$q2 = q2 + (qa \cdot gy - qb \cdot gz + q3 \cdot gx)$$

$$q3 = q3 + (qa \cdot gz + qb \cdot gy - qc \cdot gx)$$

(4.9)

**Normalizace kvaternionů:**

$$q0 = q0 \cdot \sqrt{q0^2 + q1^2 + q2^2 + q3^2}$$

$$q1 = q1 \cdot \sqrt{q0^2 + q1^2 + q2^2 + q3^2}$$

$$q2 = q2 \cdot \sqrt{q0^2 + q1^2 + q2^2 + q3^2}$$

$$q3 = q3 \cdot \sqrt{q0^2 + q1^2 + q2^2 + q3^2}$$

(4.10)

Jak je vidět z výše uvedených vzorců (4.1) až (4.10) k výpočtu kvaternionů je potřeba znát data ze všech tří čidel (všechny tři osy daného čidla), ale v programu je ošetřena i situace, kdy přestane fungovat magnetometr, akcelerometr, gyroskop nebo dvě ze tří čidel. Pokud je nefunkční magnetometr nevykonají se rovnice (4.2) až (4.5) a výpočet chyby je počítán pouze z dat akcelerometru podle vztahu (4.11). Přestane-li čidlo pracovat se pozná podle přijatých dat. Jsou-li nulová je čidlo vadné, protože gravitační pole má hodnotu 1G v některé z os (horizont je kolmý k některé ose) nebo rozloženou hodnotu do několika os (pak musí platit:  $I = \sqrt{ax^2 + ay^2 + az^2}$ ). Podobně je tomu i u magnetometru. Gyroskop má při nulové rotaci hodnotu 0 na všech osách, ale pokud ani při pohybu nedochází ke změně, jsou tyto nulová data „přebity“ hodnotami akcelerometru a magnetometru.

**Chyba mezi odhadovaným a naměřeným gravitačním polem:**

$$halfex = ay \cdot halfvz - az \cdot halfvy$$

$$halfey = az \cdot halfvx - ax \cdot halfvz$$

$$halfez = ax \cdot halfvy - ay \cdot halfvx$$

(4.11)

### 5.3.3 Kalmanův filtr

Kalmanův filtr je podobný nástroj pro filtraci dat jako je např. aritmetický průměr naměřených dat nebo plovoucí průměr. Výhodou tohoto filtru je ale v tom, že umí nejen data filtrovat (vyhladit), ale také předurčit, jaká data by měly přijít z dalšího měření a porovnat je se skutečnými data, které přijdou. Tímto krokem se zjistí, jestli data nejsou silně zarušena nebo zcela špatná, a pokud ano jsou tyto nová data upraveny podle předchozích dat. Nevýhodou Kalmanova filtru je v tom, že je nutné jej vhodně nastavit. Při špatném nastavení může filtrovat málo (nevyhladí špičky, neodstraní šum) nebo naopak filtruje až moc a nejsou zachyceny rychlé změny důležitých dat (například změna rotace). Pro správné nastavení je potřeba udělat několik pokusů a experimentální metodou tento filtr nastavit.

**Filtrace a predikce je popsána následujícím postupem:**

Pokud předpokládáme diskretní lineární systém ve tvaru

$$x_{k+1} = Ax_k + Bu_k + w_k$$

$$y_k = Cx_k + z_k$$

(4.12)

Kde  $A$  je matice dynamiky systému,  $B$  je matice vstupní,  $C$  je matice výstupní,  $x$  je vektor stavu,  $y$  je výstup. Proměnná  $w$  se nazývá systémový šum a  $z$  se nazývá šum měření.

Kalmanův filtr nejdříve spočítá odhad stavu  $\hat{x}_{k+1}$  v čase  $k$  v budoucím čase  $k+1$ .

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k$$

(4.13)

V čase  $k+1$  je možné provést porovnání s předchozího odhadu stavu se skutečným stavem a zjistit tak chybu odhadu.

$$\tilde{x}_{k+1} = x_{k+1} - \hat{x}_k$$

(4.14)

Z této chyby lze vypočítat kovariační matici

$$P_k = E[\tilde{y}_{k+1}\tilde{y}_{k+1}^T] = CP_kC^T + S_z$$

(4.15)



Je-li vypočítán odhad stavu, vypočítáme tzv. apriorní odhad výstupu

$$\hat{y}_{k+1} = C\hat{x}_k. \quad (4.16)$$

Opět provedeme srovnání předchozího stavu odhadu výstupu se skutečným stavem v čase  $k+1$ , čímž se zjistí chyba odhadu

$$\tilde{y}_{k+1} = y_{k+1} - C\hat{x}_k \quad (4.17)$$

Z této chyby lze vypočítat kovariační matici

$$R_k = E[\tilde{y}_{k+1}\tilde{y}_{k+1}^T] = CP_kC^T + S_z \quad (4.18)$$

Po vypočítání chyby odhadu lze korigovat odhad stavu pomocí matice  $K_k$ , která minimalizuje kovariační chyby

$$\begin{aligned} \tilde{x}_{k+1} &= x_{k+1} - \hat{x}_k \\ \hat{x}_{k+1} &= (A\hat{x}_k + Bu_k) + K_k(y_{k+1} - C\hat{x}_k) \end{aligned} \quad (4.19)$$

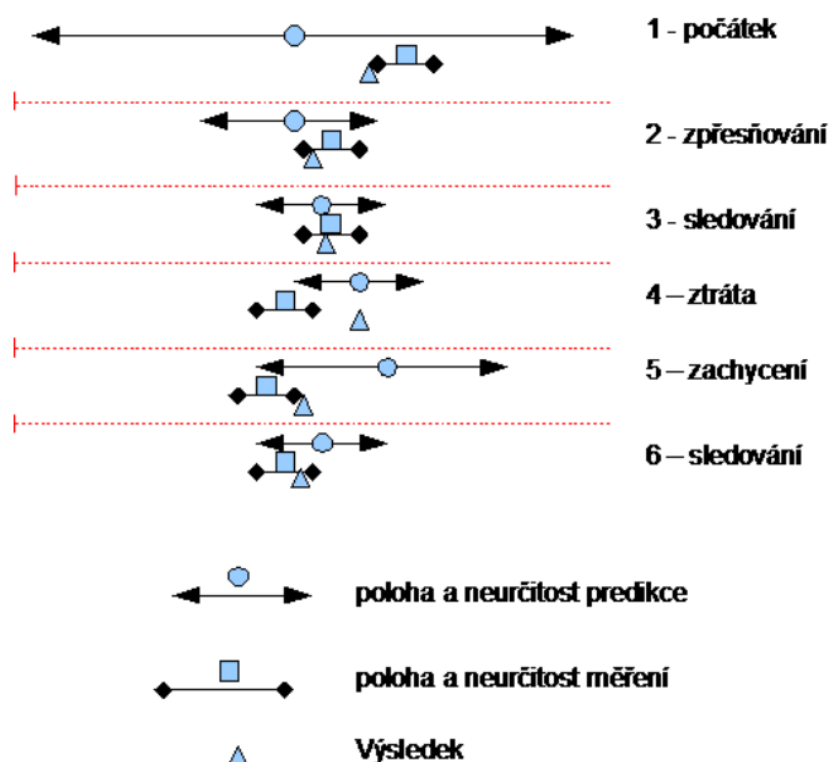
Matice  $K_k$  se vypočítá pomocí vztahu

$$K_k = AP_kC^T(CP_kC^T + S_z)^{-1} \quad (4.20)$$

Kovariační matice na následující stav se vypočítá následovně

$$P_{k+1} = AP_kA^T + S_w - AP_kC^TS_z^{-1}CP_kA^T \quad (4.21)$$

Všechny výše uvedené výrazy lze popsat obrázkem 27, na kterém je graficky znázorněno, jak takový Kalmanův filtr pracuje.



Obrázek 27: Funkce Kalmanova filtru

### Stručný popis obrázku 27, respektive funkce Kalmanova filtru:

**Počátek** – Určí nejlepší předpokládanou polohu a její neurčitost stanovíme podle toho, jak jí věříme. Dále hledáme aktuální polohu v rozmezích neurčitosti. Výsledná poloha se určí jako kombinace predikce a měření.

**Zpřesnění** – Na základě minulých hodnot a stanoveného modelu se vypočítá predikce. Neurčitost je dána vývojem minulé neurčitosti podle modelu. Proběhne měření a vypočet výsledné polohy

**Sledování** – predikce se při delším sledování zpřesňuje.

**Ztráta** – Ke ztrátě dochází při nečekané změně polohy mimo popis modelu a jeho parametrů. Následkem ztráty se zvýší neurčitost predikce.

**Zachycení** – Po ztrátě je potřeba opět zvětšit okruh, ve kterém hledáme měřenou hodnotu. Zachycení může trvat i několik kroků, než se najde měřená hodnota a filtr se opět ustálí. Až dojde k zachycení opět se zpřesňují neurčitosti a tím i poloha. Následně se proces opakuje Sledováním, které sleduje porovnává měřenou a predikovanou hodnotu.

### 5.3.4 Výpočet rotace pomocí kvaternionů

Pro rotace kolem os XYZ se používají anglické výrazy Pitch (Y), Roll (X) a Yaw (Z). Jejich výpočet se ale podle různé literatury mění, ale výsledek je téměř stejný. Ukažme si dvě možné varianty a to výpočet Pitch pomocí *asin* (4.22) nebo *atan* (4.23) a (4.24)

Výpočet rotace pomocí funkcí arcustangenty2 a arcussinus

$$\begin{aligned} Roll &= \tan^{-1}(2 \cdot q_1 \cdot q_2 - 2 \cdot q_0 \cdot q_3, 2 \cdot q_0 \cdot q_0 + 2 \cdot q_1 \cdot q_1 - 1) \\ Pitch &= -\sin^{-1}(2 \cdot q_1 \cdot q_3 + 2 \cdot q_0 \cdot q_2) \\ Yaw &= \tan^{-1}(2 \cdot q_2 \cdot q_3 - 2 \cdot q_0 \cdot q_1, 2 \cdot q_0 \cdot q_0 + 2 \cdot q_3 \cdot q_3 - 1) \end{aligned} \quad (4.22)$$

Výpočet rotace pomocí funkcí arcustangent2 a arcustangent

$$\begin{aligned} gx &= 2(q_1 \cdot q_3 - q_0 \cdot q_2) \\ gy &= 2(q_0 \cdot q_1 + q_2 \cdot q_3) \\ gz &= q_0 \cdot q_0 - q_1 \cdot q_1 - q_2 \cdot q_2 + q_3 \cdot q_3 \end{aligned} \quad (4.23)$$

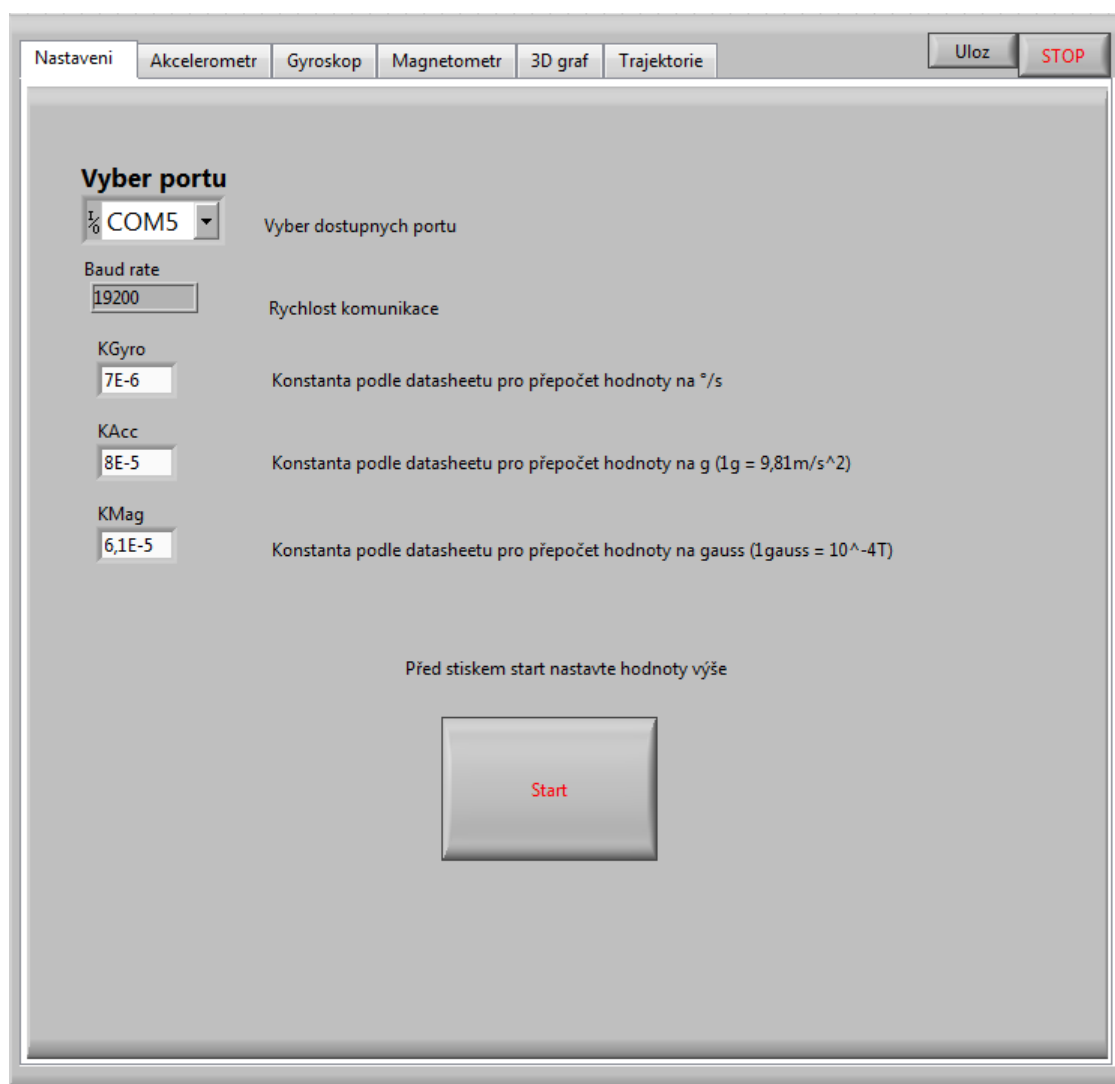
$$\begin{aligned} Roll &= \tan^{-1}(2 \cdot q_1 \cdot q_2 - 2 \cdot q_0 \cdot q_3, 2 \cdot q_0 \cdot q_0 + 2 \cdot q_1 \cdot q_1 - 1) \\ Pitch &= \tan^{-1}\left(2(q_1 \cdot q_3 - q_0 \cdot q_2) / \sqrt{gy^2 + gz^2}\right) \\ Yaw &= \tan^{-1}\left(2(q_0 \cdot q_1 + q_2 \cdot q_3) / \sqrt{gx^2 + gz^2}\right) \end{aligned} \quad (4.24)$$

## 5.4 Čelní panel vizualizace

Čelní panel byl vytvořen jako okno přepínající se pomocí záložek. Aplikace má tedy stále stejný rozměr a záložkami se jen přepínají okna, které nás zajímají. Obrázky níže jen ukazují, jak tato vizualizace vypadá. Celý program vizualizace je uveden v příloze 6.

### 5.4.1 Záložka Nastavení

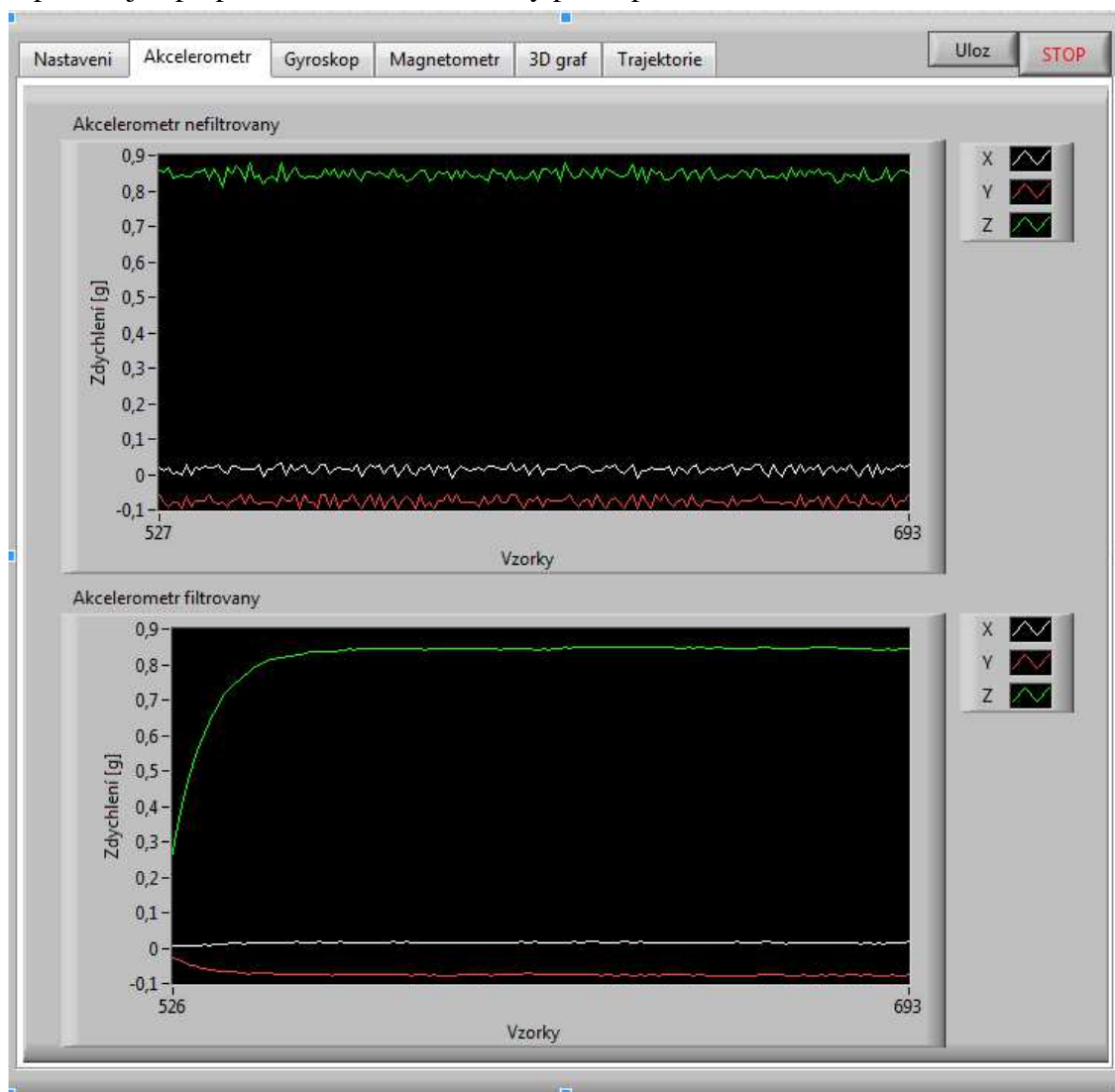
Jedná se o úvodní obrazovku a má za úkol donutit uživatele nastavit hodnoty do příslušných kolonek. Hodnoty jsou sice přednastavené, ale pro každý počítač mohou být odlišné. Pro spuštění aplikace slouží tlačítko *Start*. Tlačítkem *Ulož* se dosavadně neměřené hodnoty uloží do textového souboru a tlačítkem *STOP* se celé aplikace zastaví.



Obrázek 28: Záložka Nastavení

#### 5.4.2 Záložka Akcelerometr, Gyroskop a Magnetometr

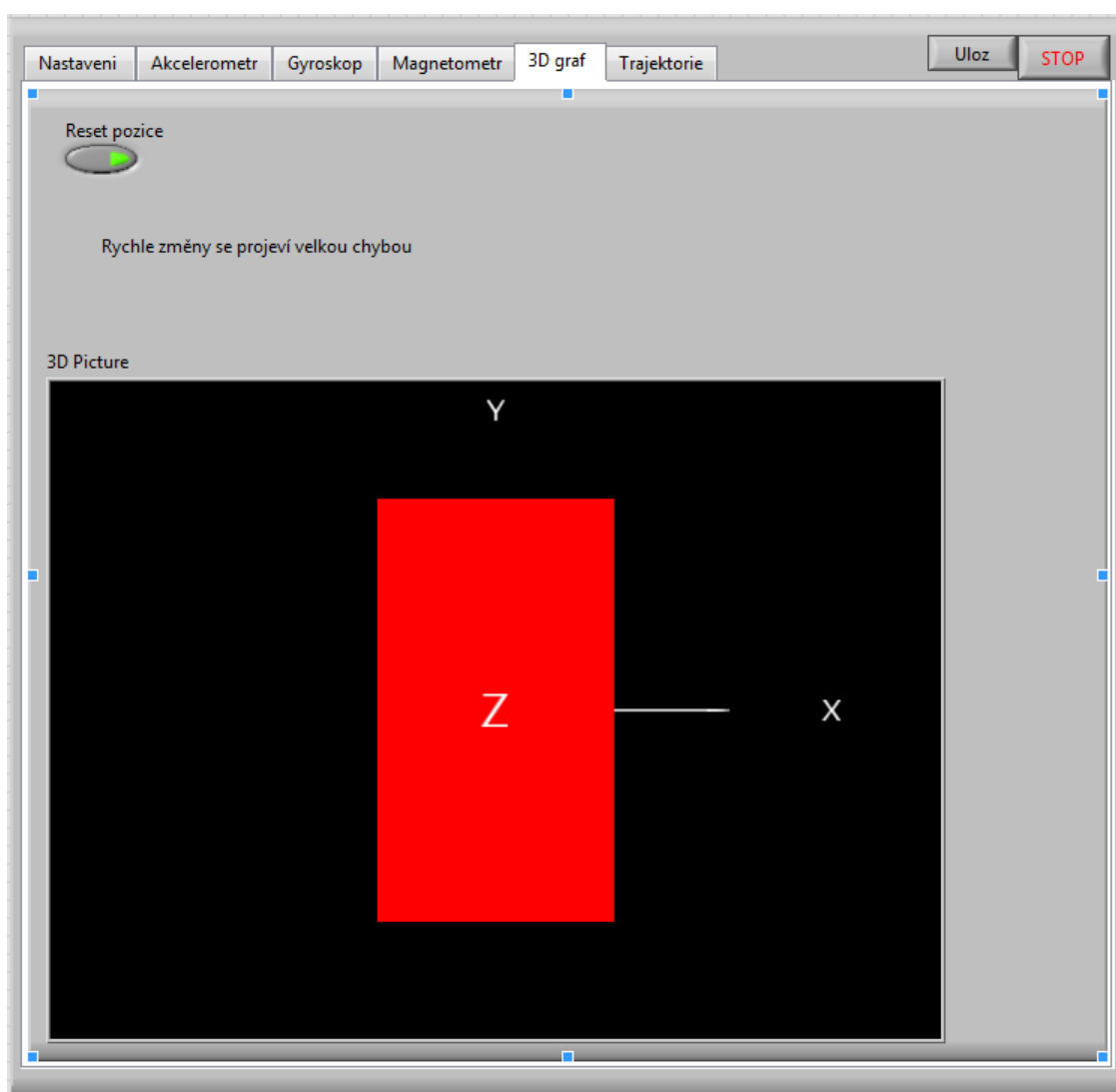
Tyto záložky zobrazují aktuální hodnoty naměřených na jednotlivých osách jednotlivými čidly. Jde o závislost hodnoty na čase respektive na počtu naměřených vzorků. Hodnoty sice přicházejí každých 20ms, ale je vhodné nezobrazovat časovou osu, protože některé hodnoty nedorazí a pak je čas mezi dvěma aktuálními daty 40ms i víc, podle toho kolik dat se „ztratí“. Horní graf představuje surová data vynásobená konstantou příslušné jednotky. V těchto datech lze vidět libovolný zákmit změny polohy či lehkých otřesů. Dolní graf představuje vyfiltrovaný průběh pomocí Kalmanova filtru. Vyfiltrovaný průběh je téměř hladký bez větších špiček (zákmitů). Pro každé čidlo bylo potřeba nastavit jiné počáteční hodnoty Kalmanova filtru a to vždy experimentální metodou. Na začátku dolního grafu je vidět, jak tento filtr funguje po zapnutí a jak přepočítává aktuální hodnoty podle predikce a korekce.



Obrázek 29: Záložka Akcelerometr

### 5.4.3 Záložka 3D graf

Slouží pro demonstraci natočení objektu v prostoru. Reaguje na pohyb modulu, ať už modul rotuje na místě nebo je v pohybu a rotuje při tom. Při použití kvaternionů není výsledek rotace natolik ovlivněn pohybem modulu. I přes úpravu dat pomocí filtru a odečítání offsetu dat, je výpočet silně ovlivněn chybou, která se objevuje při rychlých změnách pohybu modulu. To je dáno nastavením filtru, který ovšem musí být zařazen do procesu, protože jinak je chyba velká i při pomalých změnách. Dalším problémem je osa Z, která není v jedné rovině v rámci všech čidel. Čidla jsou rozmístěna po DPS a zde dochází k drobným výchyilkám v datech, což vede k samovolnému natáčení ve vizualizaci i když je měřicí modul v klidu. Dále tyto výchyilky ovlivňují 3D graf i při rotaci, kde se vzájemná rafice kolem více os přenesse na velkou chybu. Pro odstranění těchto chyb bylo přidáno tlačítko *Reset pozice*, které vrátí polohu obrázku na počáteční stav, který je na obrázku 30. Vytvořené video s funkcí 3D rotace objektu naleznete v příloze 8.



Obrázek 30: Záložka 3D graf

#### 5.4.4 Zálůžka Trajektorie

Je to 3D graf vykreslující dráhu pohybu objektu v prostoru. Pro výpočet dráhy v jednotlivých osách byly využity data rotací os (Pitch, Roll, Yaw) a hodnot akcelerometru. Akcelerometr nám dává hodnotu zrychlení v jednotlivých osách a pomocí fyzikálních vzorců dynamického zrychlení lze spočítat rychlost respektive uražená dráha objektu.

$$v = a \cdot t$$

$$s = v \cdot t = \frac{1}{2} a \cdot t^2$$

(4.25)

Kde  $a$  je akcelerace v  $[m/s^2]$ ,  $v$  je rychlost  $[m/s]$  a  $s$  je uražená dráha v  $[m]$ . Tyto vztahy platí pro statický systém. Pro dynamický systém je potřeba zavést počáteční podmínky a hodnoty  $a$ ,  $v$  a  $s$  integrovat toto naměřené zrychlení.

$$v = v_0 + \int a(t)dt = v_0 + a \cdot t$$

$$s = s_0 + \int v(t)dt = s_0 + \int (v_0(t) + a(t)(t))dt = \frac{1}{2} a \cdot t^2 + v_0 \cdot t + s_0$$

(4.26)

Protože počítáme s tím, že je modul v klidu, nastavíme počáteční podmínky na nulu. Tím se nám výpočet prvního kroku zjednoduší. V dalším kroku je ale potřeba počítat s předchozími hodnotami, které se projeví v počátečních podmínkách.

Pokud objekt při pohybu i rotuje kolem některé z os, musí se přepočítat příspěvek dráhy jednotlivých os na skutečnou hodnotu podle jednotlivých rotací.

$$s_x = \frac{s_x}{\theta} + \frac{s_x}{\psi}$$

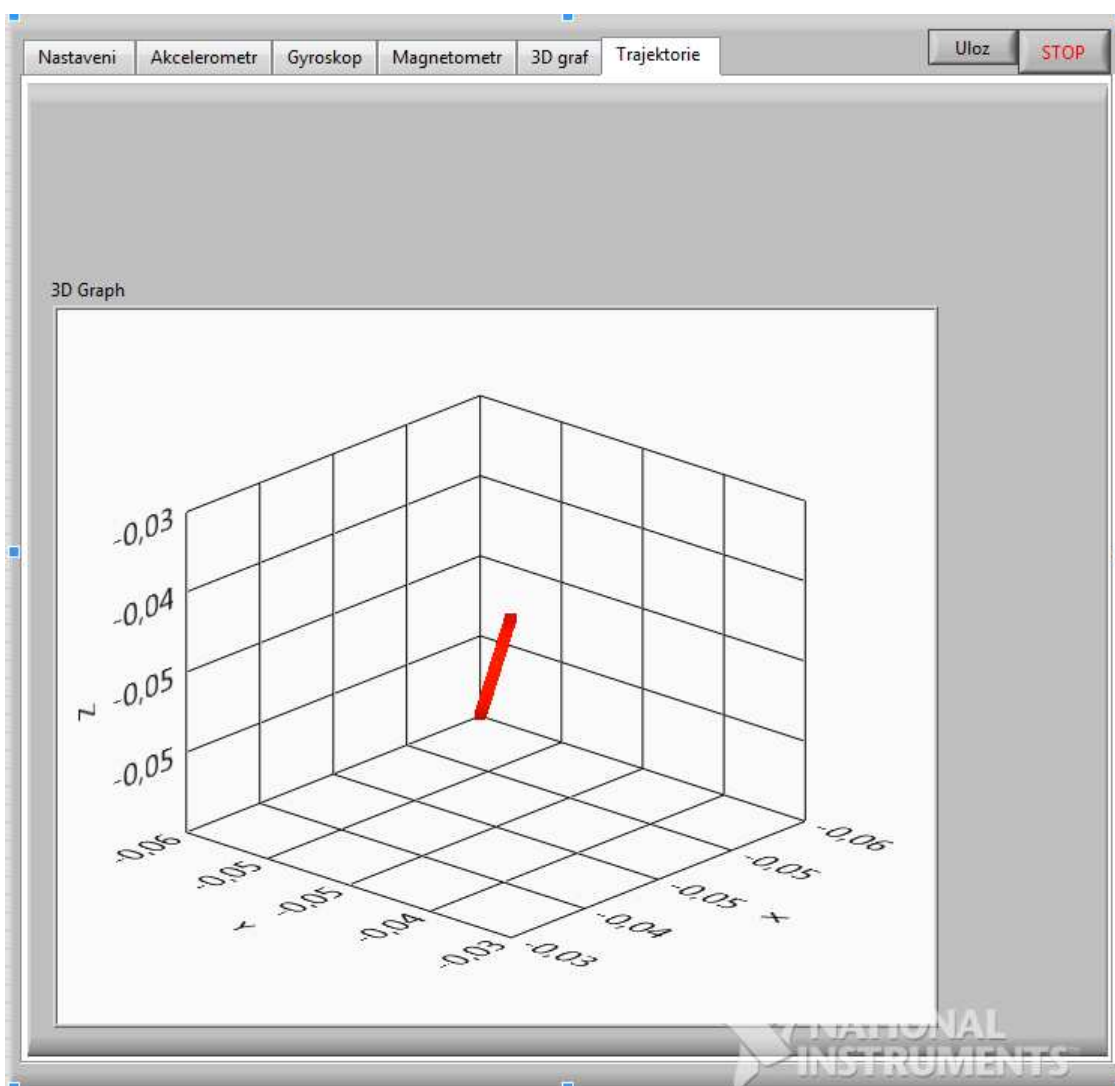
$$s_y = \frac{s_y}{\phi} + \frac{s_y}{\psi}$$

$$s_z = \frac{s_z}{\theta} + \frac{s_z}{\phi}$$

(4.27)

I když je modul v klidu zobrazuje graf drobné posuny v osách a proto se neustále vykresluje pohyb. Je to dáno stejnými chybami, které jsou popsány již výše. Vytvořené video s funkcí vypočtu trajektorie objektu naleznete v příloze 9.





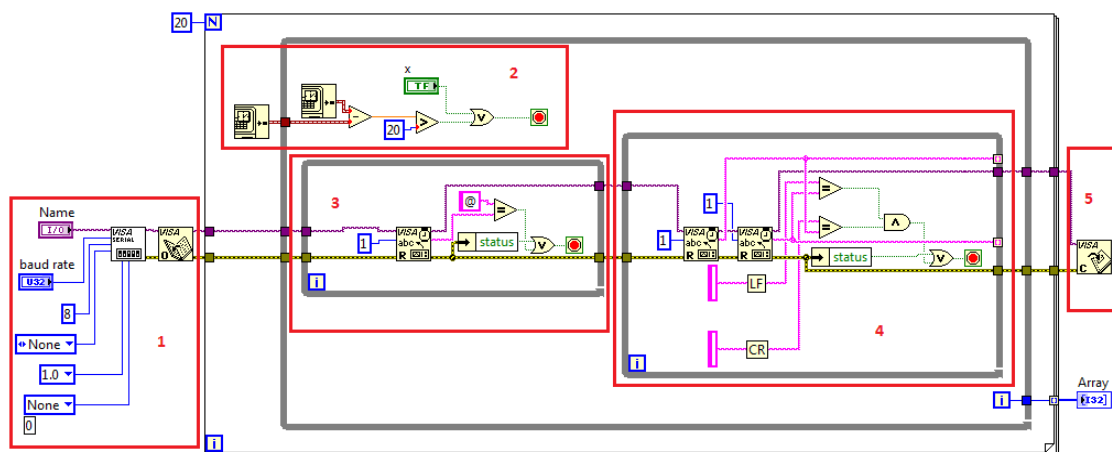
Obrázek 31: Záložka Trajektorie

## 6 Testování vytvořeného prototypu

Měřicí modul byl testován na dosah IQRF modulů, respektive množství přenesených dat a dále na rotace kolem jednotlivých os. Zobrazena a popsána je pouze rotace kolem osy Z ostatní rotace jsou v příloze 2.

### 6.1 Dosah měřicího modulu

Pro účely tohoto měření byla vytvořena měřicí aplikace taktéž v programovacím prostředí LabVIEW. Jedná se o „ořezanou“ aplikaci, která po startu pouze otevře port o příslušné rychlosti a vyčítá data v cyklu. Testovací aplikace měří vždy po dobu 20s a pro zpřesnění výsledku se tento cyklus opakuje 20x. Na obrázku 32 je tato aplikace vyobrazena a rozdělena do 5 částí, které jsou blíže popsány níže.



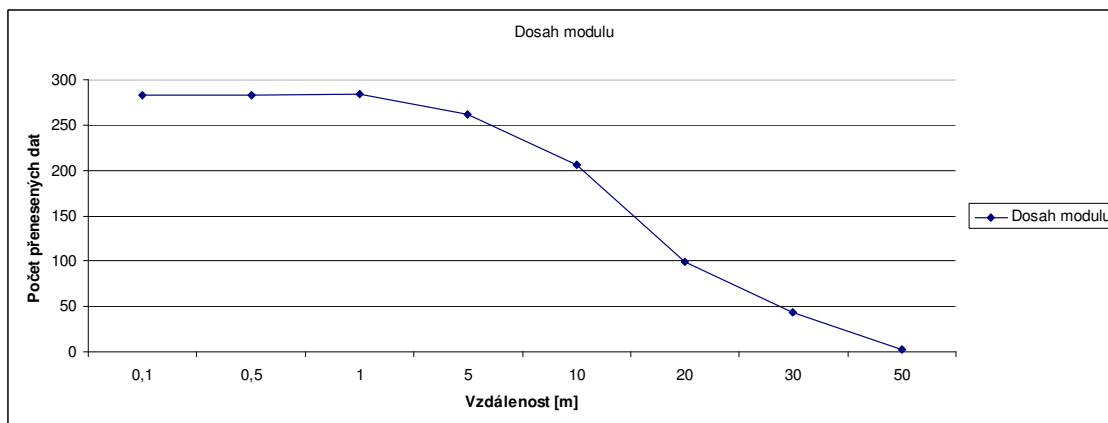
Obrázek 32: Měřicí aplikace

- 1) Inicializace UART – zde se nastaví port, rychlost a příslušné komunikační bity.
- 2) Časový rozestup – zde se měří doba běhu cyklu. Jakmile dosáhne doba měření více než 20s smyčka while se ukončí a pokračuje se cyklem for.
- 3) Hledání počátečního znaku – Standardně přicházejí znaky postupně jak jsou vysílány pomocí komunikace UART, ale je vhodné hledat v řetězci uvozující znak.
- 4) Hlavní smyčka – zde se hledají ukončovací znaky CR a LF (návrat kurzoru na začátek a odřádkování).
- 5) Ukončení komunikace a uzavření portu.

Měření bylo provedeno na volném prostranství, kde na sebe oba moduly přímo viděly. V dosahu nebyla žádná wi-fi síť a výsledné naměřené data jsou v příloze 7.

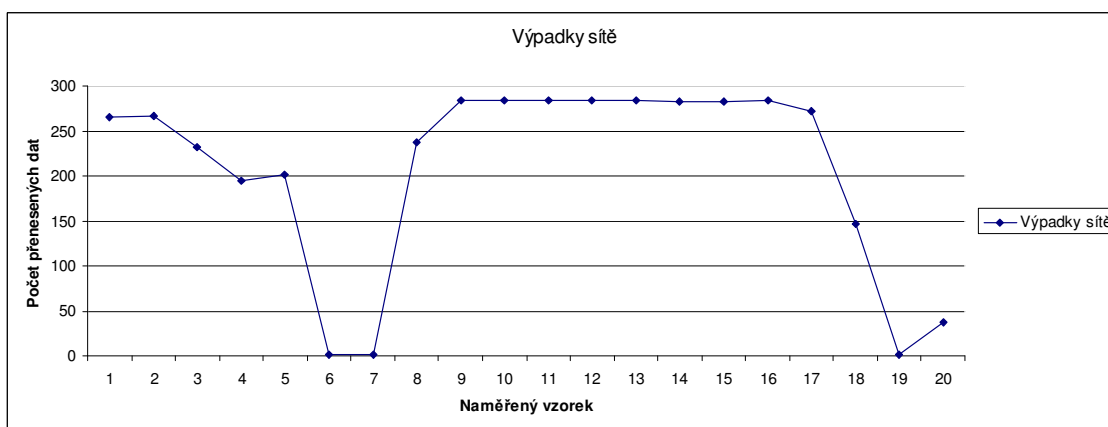
Jak je vidět na obrázku 33, tak dosah měřicího modulu je docela malý. Na vzdálenosti 50m se dá říct, že jsou moduly mimo dosah. Přenese se pouze mizivé

procento dat. Uspokojivě dopadla vzdálenost 20m, kde se dá říct, že obě zařízení spolu relativně dobře komunikují.



**Obrázek 33: Graf dosahu měřicího modulu**

Na obrázku 34 je vidět, že síť občas vypadne, a tím se přichází o většinu dat v danou chvíli. Tento fakt byl zjištěn při použití dvaceti měření pro jednu hodnotu vzdálenosti. Výsledná prostupnost dat je pak spočítána aritmetickým průměrem, kde se jednotlivé vzorky sečtou a vydělí dvaceti.



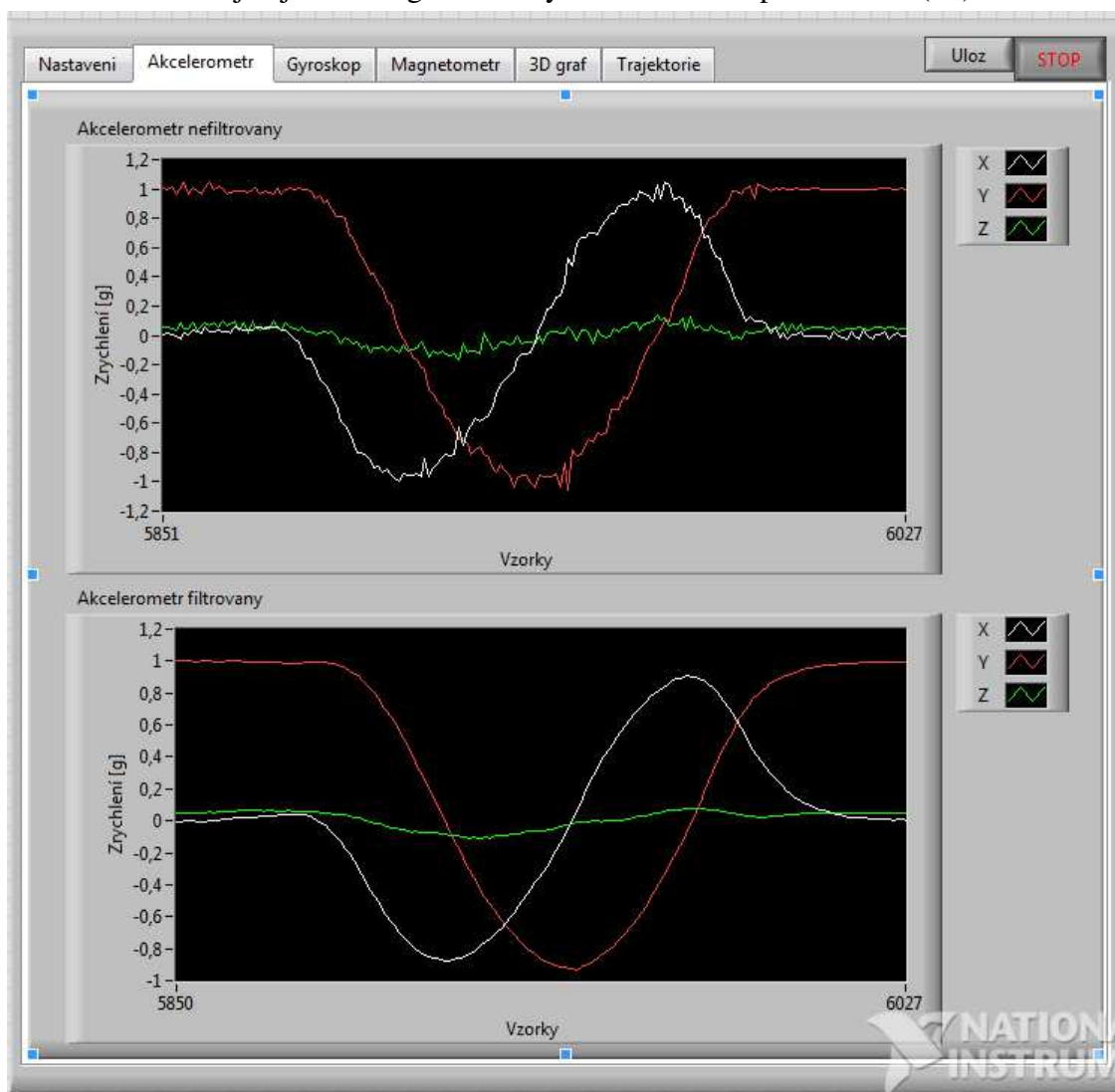
**Obrázek 34: Graf výpadku sítě**

## 6.2 Rotace kolem osy Z

Rotace kolem osy Z byla provedena pouze v rukách a tím bylo měření ovlivněno drobnými otřesy rukou. Moduly na sebe viděly a byly ve vzdálenosti cca 1m.

### 6.2.1 Akcelerometr

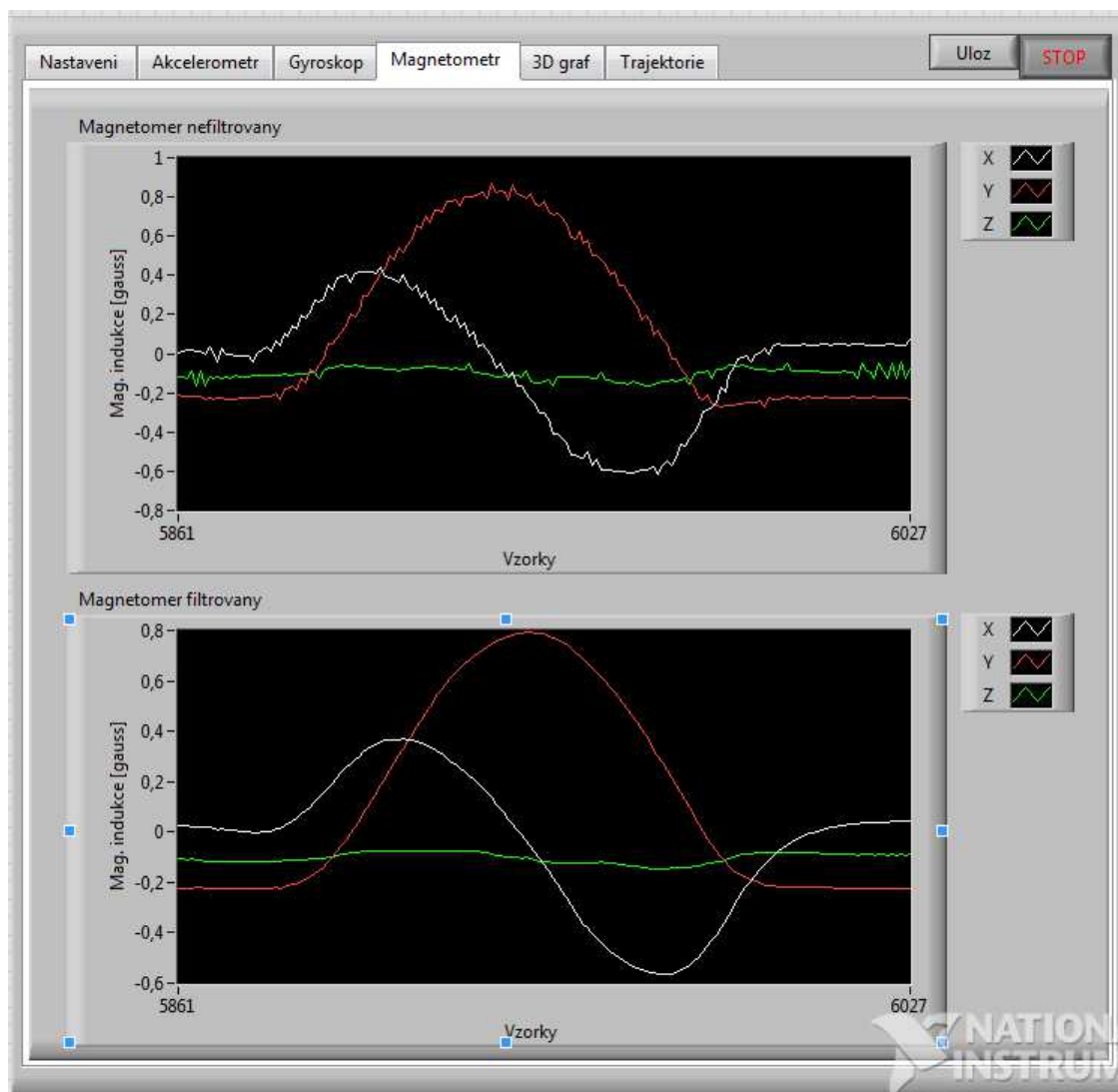
Na obrázku 34 je vidět jaký má tato rotace dopad na hodnoty akcelerometru. Osa Z se hýbe jen minimálně, ale za to osy X a Y jsou v pohybu. Y začíná funkcí kosinus a X funkcí  $-\sin$  a jak je vidět z grafu obě tyto funkce urazí přesně  $360^\circ$  ( $2\pi$ ).



Obrázek 35: Graf akcelerometru při rotaci osy Z o  $2\pi$

### 6.2.2 Magnetometr

Magnetometr vytváří velice podobný graf (obrázek 35), s tím rozdílem, že magnetické pole je jinak rozloženo v klidovém režimu. Opět je osa Z téměř nehybná a naopak osy X a Y vytvářejí ty samé funkce jako u akcelerometru, ale jsou přehozeny. Osa Z ukazovala na západ, a proto při rotaci kolem této osy se nemění magnetické pole Země. Osa X ukazovala na sever, a proto je vidět jak si při rotaci vyměňovaly hodnoty s osou Y.

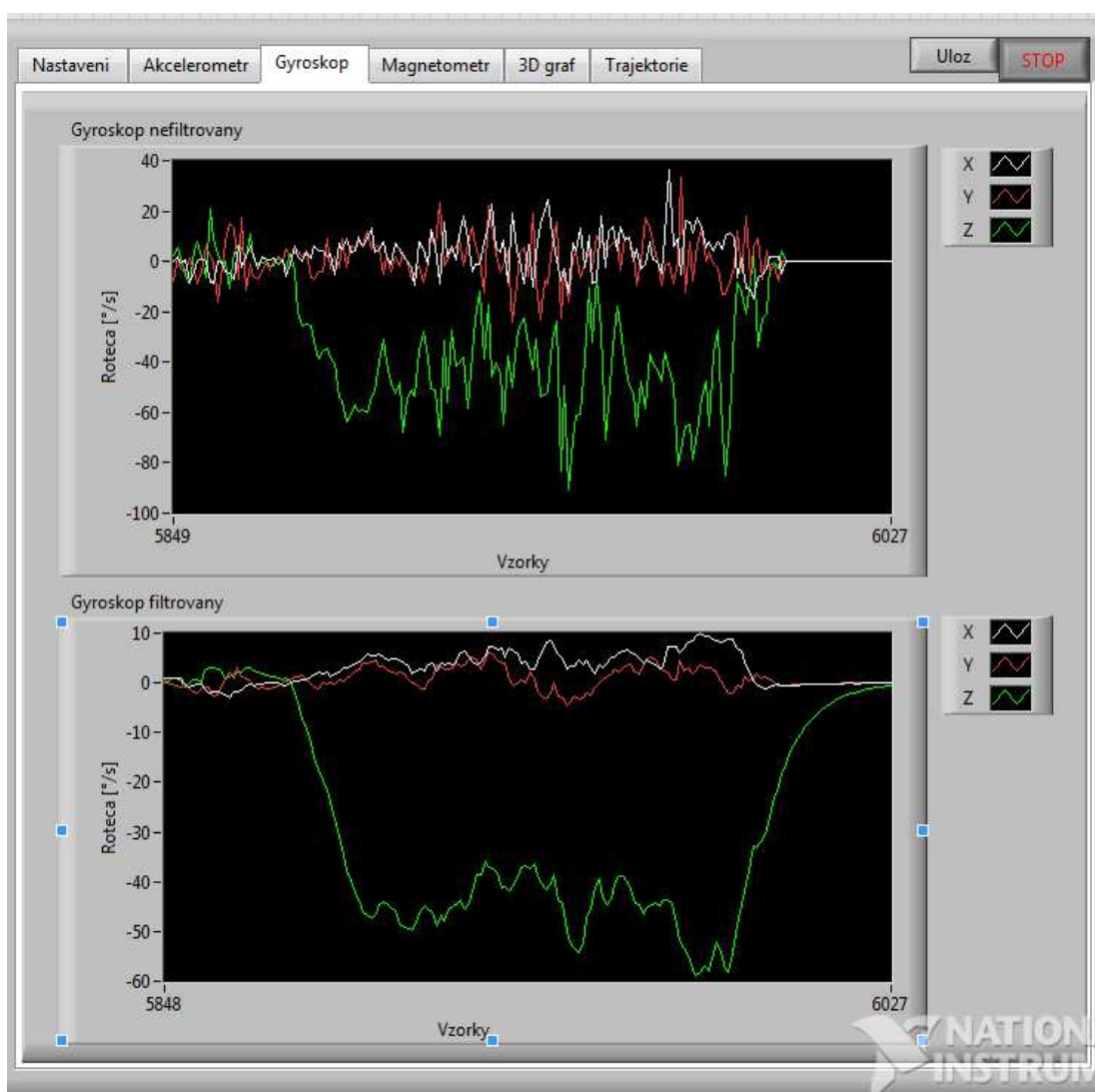


Obrázek 36: Graf magnetometru při rotaci osy Z o  $2\pi$

### 6.2.3 Gyroskop

Gyroskop ukazuje přímo rotaci a je teda na grafu vidět jak se tato rotace projeví. Nefiltrovaný graf je silně rušený třasem rukou, ale po filtraci je jasně vidět, že osy X a Y jsou téměř v klidu a osa Z napočítává celkovou rotaci kolem této osy. Zde se dostáváme do problému, jak silně nastavit filtr, aby nebyla ovlivněna správnost dat, ale zároveň, aby data byly co nejčistší.

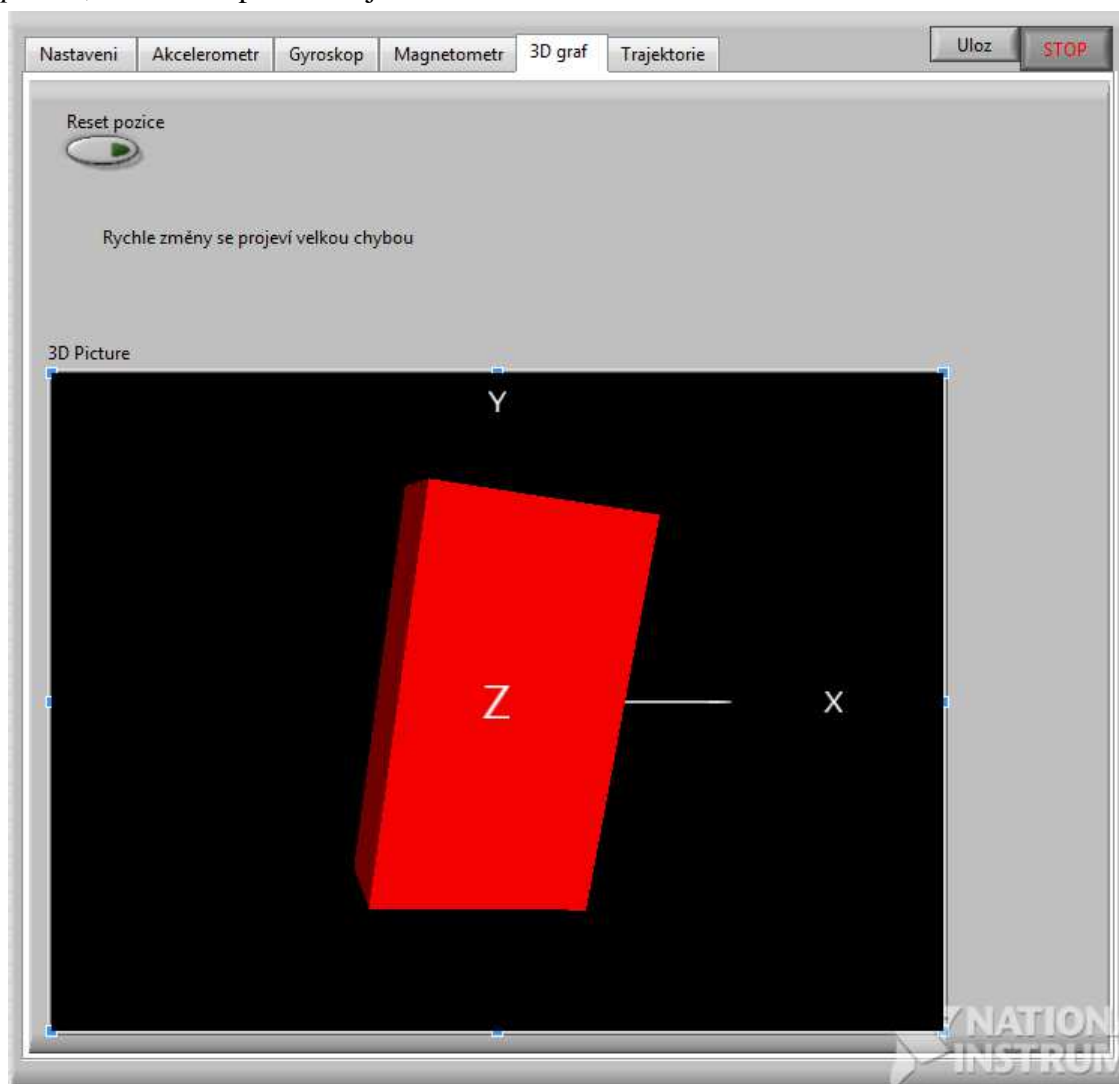
V tomto měření se jednalo o pomalé změny a je vidět, že by mohl být filtr nastaven na větší stupeň, ale při rychlejší změně se filtrace projeví více, a tím zahodíme cenná data. Proto bylo nutné vytvořit kompromis v nastavení filtru, který je vidět na obrázku 36.



Obrázek 37: Graf gyroskopu při rotaci osy Z o  $2\pi$

#### 6.2.4 3D graf rotace

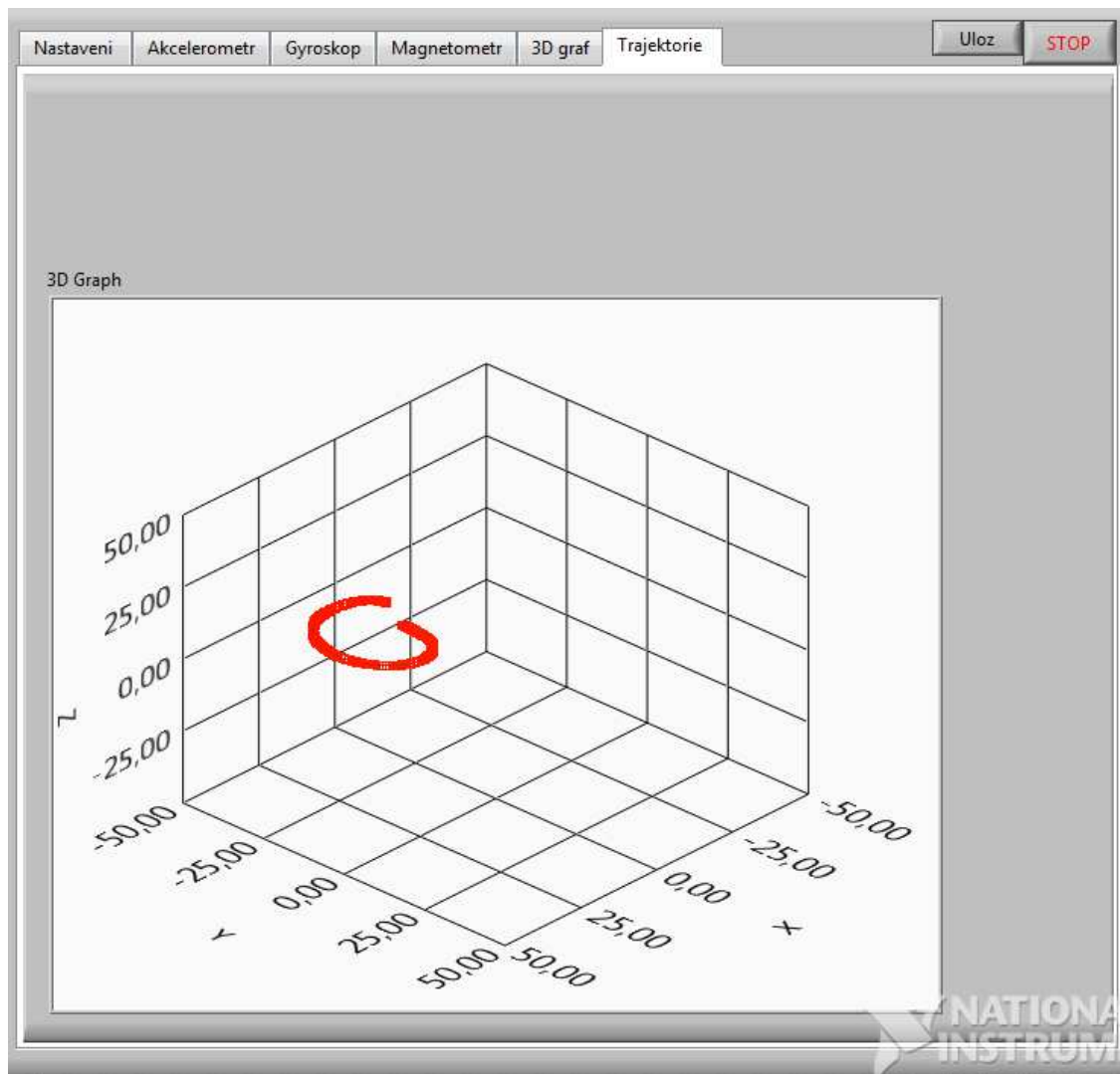
Tento graf na obrázku 37 ukazuje, jak dopadne tato rotace za použití AHRS algoritmu. Je vidět, že se i drobný třas přenesse na rotační pohyb a objekt se hýbe i v ostatních osách. Jelikož šlo o pomalý pohyb je chyba středně velká. Při středně rychlém pohybu je nastavení filtru optimální, a proto by byla chyba minimální a při rychlém pohybu je opět chyba středně velká. Extrémně rychlý pohyb nezaznamená měřicí modul, a proto dochází k velkým chybám a je nutné používat tlačítko *Reset pozice*, které vrátí polohu objektu na defaultní hodnotu.



Obrázek 38: 3D graf při rotaci osy Z o  $2\pi$

### 6.2.5 Trajektorie

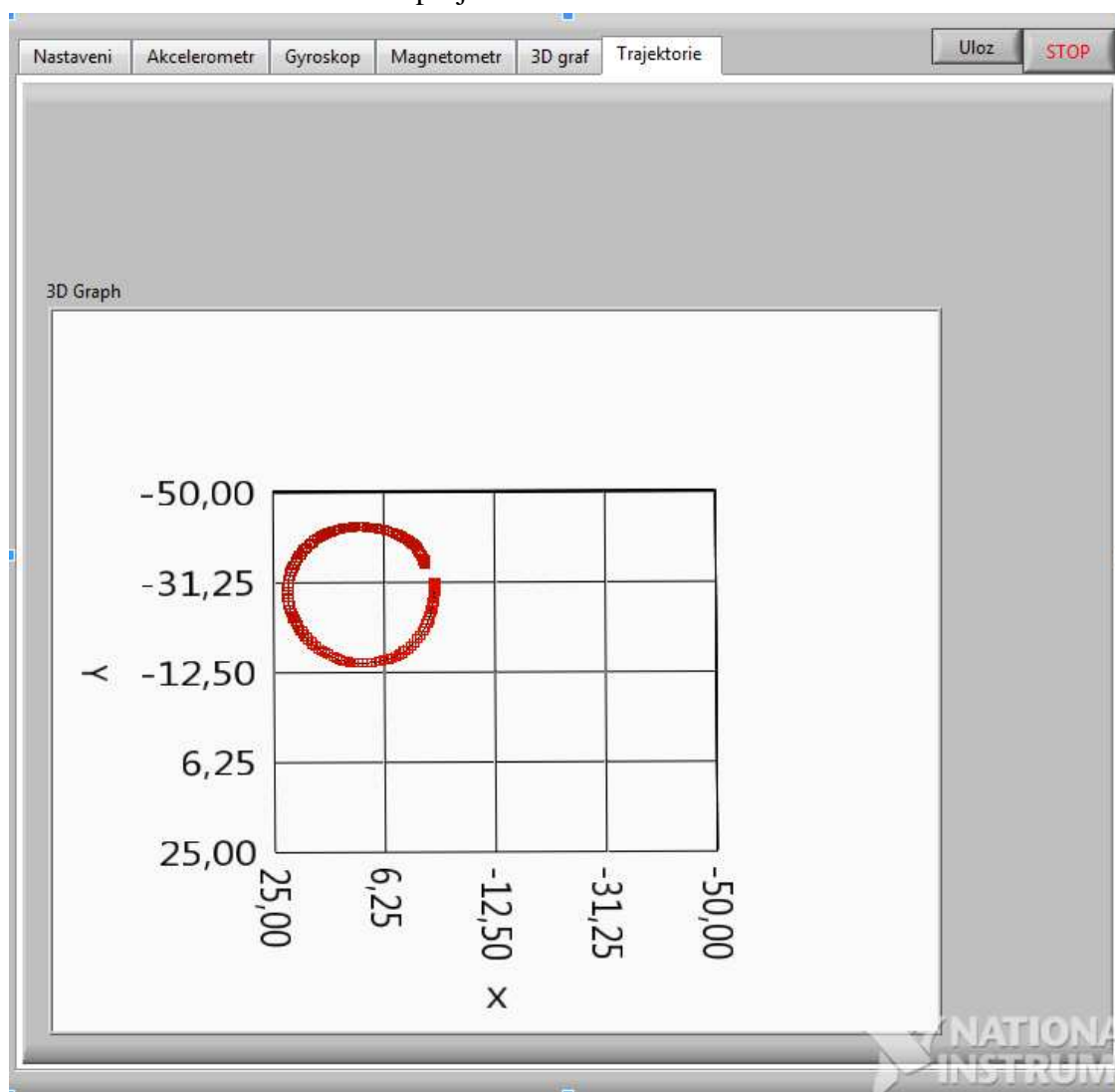
Na obrázku 38 je vidět jako rotace kolem osy Z dopadla. U grafu trajektorie musí být zapnutá funkce autoscale, aby se osy zvětšovaly podle přijatých dat a nedocházelo tak k vykreslení grafu za osy apod. Z tohoto pohledu je vidět, že se zvětšovaly osy X a Y a osa Z se zvětšovala jen minimálně. Lépe je vidět tento graf z pohledu na rovinu XY, která je na obrázku 39.



Obrázek 39: 3D trajektorie při rotaci osy Z o  $2\pi$



Zde je vidět, že osa X a Y „utíkají“ i když je objekt ještě v klidu a následně se vykreslí kruh, který vytváří rotaci osy Z. Kruh není dokonale kulatý, protože se do měření musejí přidat drobné otřesy modulu a fakt, že čidla nejsou uprostřed všech tří os a i tato minimální vzdálenost se projeví.



Obrázek 40: 3D trajektorie při rotaci osy Z o  $2\pi$  (pohled na rovinu XY)

## 7 Zhodnocení dosažených výsledků

Při výrobě HW části došlo k drobným potížím a to jak při výrobě DPS, tak při opracování krabičky. Vlivem přidání součástek do měřicího modulu se musela DPS vyrobit znovu, a proto bylo i potřeba upravit krabičku. To má za následek neestetický vzhled, který je způsoben převrtáním děr pro USB a úchyt DPS.

Modul AltIMU obsahuje barometr, ale v této práci pro něj není žádné využití, proto nebyly jeho data použita. Velký problém nastal při vytváření SW pro přijímací část. Podle datasheetu modul TR-56D umí spolehlivě komunikovat pouze pro nastavení komunikační rychlosti 19,2kb/s. Ostatní rychlosti nejsou odladěny a slouží pouze pro experimentální účely. Při vytváření přijímací části nastala potřeba toto nastavení rychlosti zvýšit. Vyšší rychlosti sice fungují, ale za cenu ztráty většinu naměřených dat. Tento fakt byl zjištěn pomocí experimentální metody, při kterém se měnilo nastavení obou komunikačních rozhraní (jak rádiové, tak sériové). Různé nastavení rychlosti sériové komunikace neovlivňoval výsledek, ale nastavení rychlosti RF komunikace měl na funkci velký vliv. Proto bylo opuštěno od možnosti připojení více měřicích stanic. Testované komunikační rychlosti byly tyto:

UART - 9,6 kb/s, 19,2 kb/s 38,4 kb/s, 62,5 kb/s

RF - 19,2kb/s, 57,6kb/s, 86,2kb/s

Vytvořená vizualizace zobrazuje několik záložek popsaných v kapitolách výše. Aktuální data akcelerometru, gyroskopu a magnetometru nám ukazují, jak funguje filtrace tvořená kalmanovým filtrem. Filtrace byla odladěna experimentální metodou pro každé čidlo zvlášť, ale u gyroskopu byla nastavena citlivost filtrace na středně rychlé pohyby, což ale vede paradoxně k větším chybám při pohybech rychlých nebo naopak velmi pomalých. AHRS algoritmus byl úspěšně implementován, ale jeho výpočet vede k drobným odchylkám od skutečných hodnot, a proto při zobrazení 3D grafu rotace se objekt pohybuje, i když tomu tak není. 3D graf rotace byl vytvořen jako obrázek rotující podle vypočtených rotací a 3D trajektorie vykresluje čáry mezi jednotlivými body pozice. Trajektorie objektu v prostoru je vypočítávána pomocí AHRS algoritmu a další pomocných rovnic, a proto je velmi náchylná na správnost naměřených dat. Při rotaci kolem jednotlivých os je tato chyba malá, ale při pohybu v prostoru vykazuje tento graf velké odchylky od správných hodnot. To se potvrdilo i při měření rotace kolem jednotlivých os, které je v kapitole výše.

Měření dosahu IQRF modulů ukázalo, že nejsou tyto moduly moc vhodné na přenos dat na větší vzdálenosti, a proto se pro tuto konkrétní úlohu nehodí. Nicméně moduly mají uplatnění v jiných odvětvích, kde jsou velkým přínosem. Mezi ně patří

např. je lze využít ke konstrukci chytrého domu, výrobě bezdrátových zvonků, rozhlasu apod.

Možností zlepšení celkového výsledku této práce je například v požití externí antény pro dosažení většího dosahu, případně použití jiného komunikačního modulu. V této práci použití obou možností nepřipadala v úvahu, protože jsem pracoval s moduly TR-56D, které jsou podle zadání IQRF moduly a zároveň mají vytvořenou anténu na DPS. Podobně lze vylepšit i prostupnost dat, kde by bylo možné použít jiný komunikační modul, který lze nastavit na větší komunikační rychlost. Jedna z poslední možnosti, jak celé zařízení vylepšit, je vytvořit lepší algoritmus pro odstranění gravitační složky akcelerometru, která v případě výpočtu trajektorie zanáší největší chybu.

## 8 Seznam literatury

[1] SEFLOVA, Petra et al. IQRF wireless technology utilizing IQMESH protocol. In: *35th International Conference on Telecommunications and Signal Processing, TSP 2012*. Prague, Jul. 2012, p. 101-104. ISBN 978-1-4673-1117-5. doi: 10.1109/TSP.2012.6256261.

[2] MICRORISC s.r.o. *Complete technology for MESH networks* [online]. 2013 [cit.2014-10-16]. Dostupné z: <http://iqrf.org/weben/index.php>.

[3] VRBA, Kamil et al. *Konstrukce elektronických zařízení*. [Online]. Brno, 2013 [cit. 2015-1-10]. Vysoké učení technické v Brně, 2013. s. 251. ISBN 978-80-214-4719-6. Dostupné také z: [http://homen.vsb.cz/~vac174/vyuka/KZPE/Studijni\\_opory/Konstrukce\\_elektronickych\\_zarizeni.pdf](http://homen.vsb.cz/~vac174/vyuka/KZPE/Studijni_opory/Konstrukce_elektronickych_zarizeni.pdf)

[4] GANG, Sun et al. Implementing quaternion based AHRS on a MEMS multisensor hardware platform. In: *International Global Navigation Satellite Systems Society IGNSS Symposium 2013*. Qld Australia, Jul. 2013 [cit. 2015-1-10].

[5] MARCARD, Timo Von. *Design and implementation of an attitude estimation system to control orthopedic components*. [Online]. Göteborg, Sweden 2010 [cit. 2015-1-10]. Dostupné také z: <http://publications.lib.chalmers.se/records/fulltext/125985.pdf>

[6] WIKIPEDIA. *Nutation* [Online]. 16.11.2010, [cit. 2015-1-10]. Dostupné z: <http://en.wikipedia.org/wiki/Nutation?oldid=397185783>

[7] HÁJEK, L., HAMBÁLEK, T. *Gyroskopy* [Online]. [cit. 2015-1-10]. Dostupné z: <http://fyzsem.fjfi.cvut.cz/2010-2011/Zima10/proc/gyroskopy.pdf>

[8] *Acclerometer & Gyro Tutorial* [Online]. [cit. 2015-1-10]. Dostupné z: <http://www.instructables.com/id/Accelerometer-Gyro-Tutorial/>

[9] GÁBRLÍK, Petr. *Test akcelerometru Freescale MMA7260QT* [Online]. 13.5.2009 [cit. 2015-1-10]. Dostupné z: [http://pandatron.cz/?776&test\\_akcelerometru\\_freescale\\_mma7260qt](http://pandatron.cz/?776&test_akcelerometru_freescale_mma7260qt)

[10] *Co je a k čemu slouží magnetometr* [Online]. 24.7.2008 [cit. 2015-1-10]. Dostupné z: <http://www.enviweb.cz/clanek/geologie/71131/co-je-a-k-cemu-slouzi-magnetometr>

[11] WIKIPEDIA. *Magnetometr* [Online]. 13.2.2014 [cit. 2015-1-10]. Dostupné z: <http://cs.wikipedia.org/wiki/Magnetometr>

[12] WIKIPEDIA. *Barometr* [Online]. 3.7.2014 [cit. 2015-1-10]. Dostupné z: <http://cs.wikipedia.org/wiki/Barometr>

[13] *TR-56D Series* [Online]. [cit. 2015-1-10]. Dostupné z: <http://iqrf.org/weben/index.php?sekce=products&id=tr-56d&ot=transceivers&ot2=tr-56d>

[14] *AltIMU-10 v4 Gyro, Accelerometer, Compass and Altimeter (L3GD20H, LSM303D and LPS25H Carrier)* [Online]. [cit. 2015-1-10]. Dostupné z: <https://www.pololu.com/product/2470>

[15] OLEJÁR, Martin. *Stručný popis sběrnice I2C a její praktické využití k připojení externí eeprom 24LC256 k mikrokontroléru PIC16F877* [Online]. 20.5.2000 [cit. 2015-1-10]. Dostupné z: <http://www.hw.cz/navrh-obvodu/strucny-popis-sbernice-i2c-a-jeji-prakticke-vyuziti-k-pripojeni-externi-eeprom-24lc256>

[16] CHMELAR, Pavel. *Stabilizace polohy létajícího objektu pomocí neinertních senzorů* [Online]. 2011 [cit. 2015-1-10]. Dostupné z: [https://dspace.upce.cz/bitstream/10195/39401/1/ChmelarP\\_StabilizacePolohy\\_PR\\_2011.pdf](https://dspace.upce.cz/bitstream/10195/39401/1/ChmelarP_StabilizacePolohy_PR_2011.pdf)

[17] *Optické senzory polohy* [Online]. [cit. 2015-1-10]. Dostupné z: [http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A3B38SME/Opticky\\_senzor\\_polohy\\_10a.pdf](http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A3B38SME/Opticky_senzor_polohy_10a.pdf)

[18] BOHATÝ, Jan. *Zpracování dat z GPS modulu* [Online]. 2013 [cit. 2015-1-10]. Dostupné z: <https://otik.uk.zcu.cz/bitstream/handle/11025/7417/Zpracovani%20dat%20z%20GPS%20modulu.pdf?sequence=1>

[19] MICROCHIP. *MCP73831/2* [Online]. [cit. 2015-1-10]. Dostupné z: <https://www.sparkfun.com/datasheets/Prototyping/Batteries/MCP73831T.pdf>

[20] IQRF. *Technology for wireless* [Online]. [cit. 2015-1-10]. Dostupné z: <http://iqrf.org/weben/>

[21] NI [Online]. [cit 2015-1-4]. Dostupné z: <http://czech.ni.com/>

[22] SLINTÁK, Vlastimil. *Konverze mezi 5V a 3,3V logikou* [Online]. 5.10.2011[cit 2015-23-4]. Dostupné z: <http://uart.cz/253/konverze-mezi-5v-a-3v-logikou/>

[23] OLMR, Vít. *Sériová linka RS232* [Online]. 12.12.2005[cit 2015-23-4]. Dostupné z: <http://www.hw.cz/rozhrani/hw-server-predstavuje-seriova-linka-rs-232.html>

[24] *Kalmanův filtr* [Online]. 17.10.2008[cit 2015-27-4]. Dostupné z: [http://www.uamt.feec.vutbr.cz/~richter/vyuka/0910\\_mpov/tmp/kalman\\_filter.html.cs](http://www.uamt.feec.vutbr.cz/~richter/vyuka/0910_mpov/tmp/kalman_filter.html.cs)

[25] Pánek, David. *Kalmanův filtr* [Online]. 11.12.2012[cit 2015-27-4]. Dostupné z: [http://home.zcu.cz/~panek50/TEVS/11\\_prednaska.pdf](http://home.zcu.cz/~panek50/TEVS/11_prednaska.pdf)

## **9 Seznam příloh**

Příloha1- DPS a ostatní – Schéma, DPS a osazovací plánec měřící části

### **9.1 Přílohy pouze na CD**

Příloha2-Rotace\_os – Výsledek měření rotace kolem všech os

Příloha3-Výkres\_víka – Výkres opracovaného víka krabíčky

Příloha4-Měřící stanice – Program měřící stanice

Příloha5-Přijímač – Program přijímače

Příloha6-Vizualizace – Program vizualizace (prostředí LabVIEW)

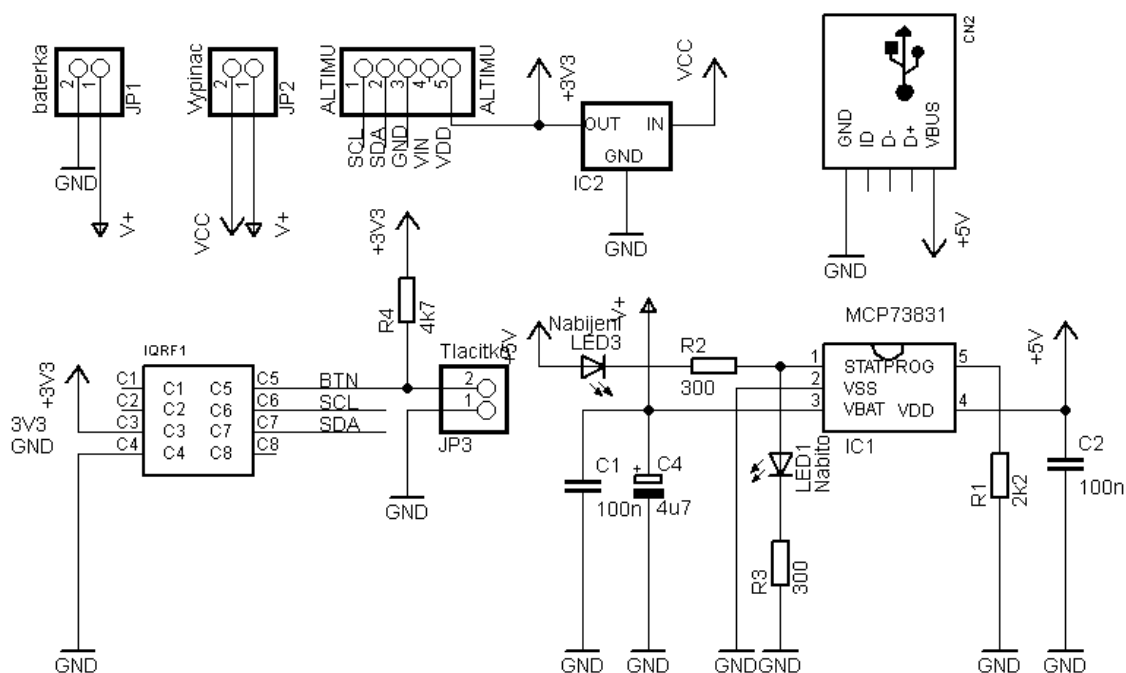
Příloha7-Měření\_dosahu – Tabulka naměřených dat

Příloha8-3D\_Graf – Video 3D rotace objektu

Příloha9-Trajektorie – Video zobrazení trajektorie při postupné rotaci kolem všech os

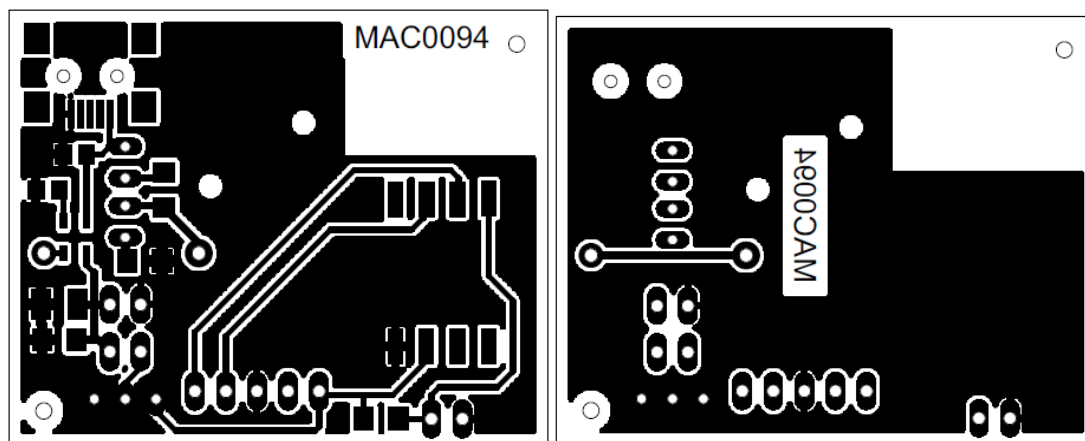
## Příloha1-DPS a ostatní

### Schéma měřicí části



Obrázek 1: Schéma propojení modulů

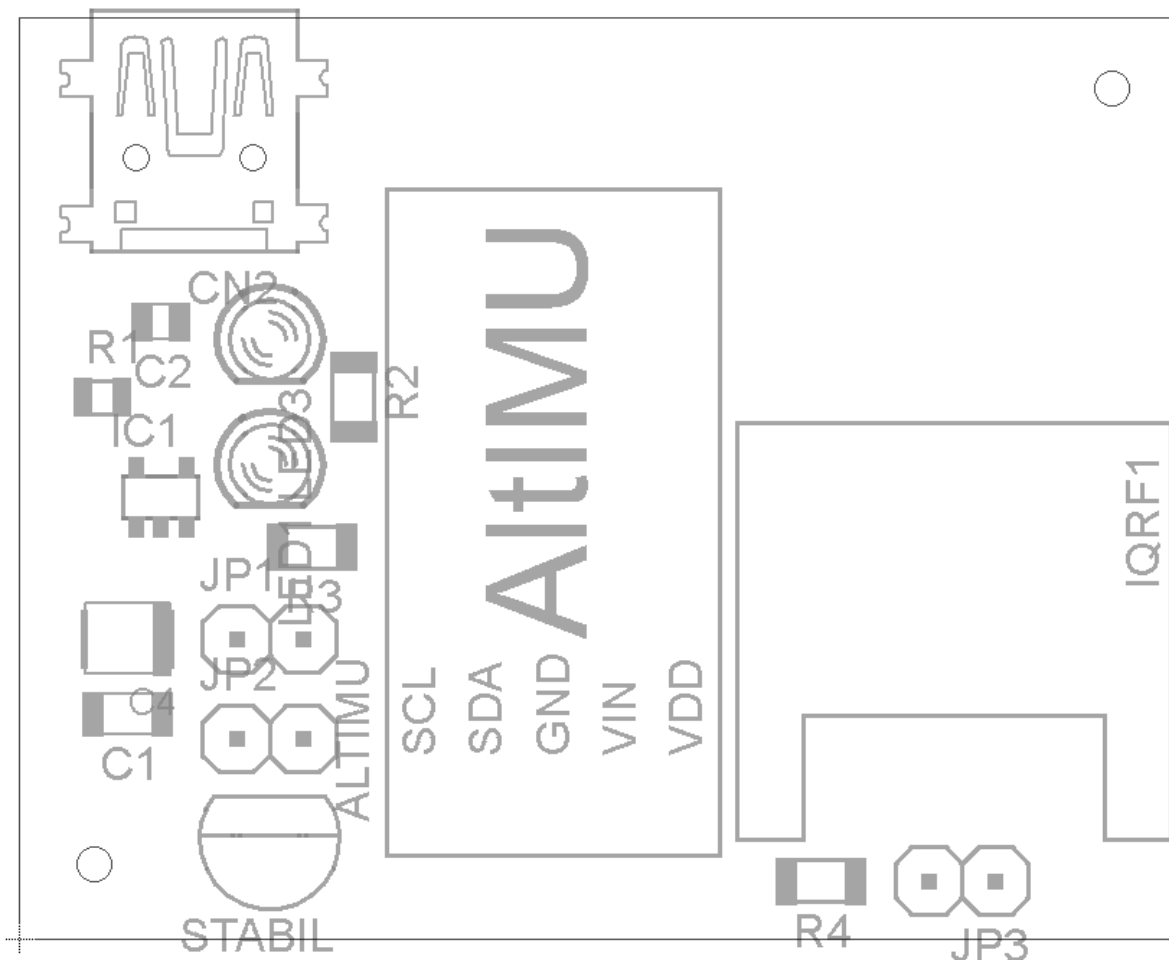
### DPS měřicí části



Obrázek 2: DPS vlevo TOP, vpravo BOTTOM



## Osazovací plánec měřicí části



Obrázek 3: Osazovací plánec strana TOP

## Seznam součástek

Tabulka 1: Seznam součástek

Partlist					
Exported from Diplomka.brd at 29.4.2015 13:45:25					
EAGLE Version 6.4.0 Copyright (c) 1988-2013 CadSoft					
Assembly variant:					
Part	Value	Package	Library	Position (mil)	Orientation
ALTIMU	ALTIMU	1X05/90	pinhead	(800 650)	R90
C1	100n	C1206	rcl	(162.5 337.5)	R180
C2	100n	C0805	rcl	(212.5 925)	R180
C4	4u7	SMC_B	rcl	(162.5 450)	R180
CN2		USB-MINIB	#jirka_2	(262.5 1212.5)	R180
IC1	MCP73831	SOT23-5L	microchip2	(212.5 662.5)	R0
IQRF1	IQRF	SCU-W0822XR-06	Zdena_2	(1400 550)	R90
JP1	baterka	1X02	pinhead	(375 450)	R0

JP2	Vypinac	1X02	pinhead	(375 300)	R0
JP3	Tlaciťko	1X02	pinhead	(1412.5 87.5)	R180
LED1	Nabito	LED3MM	led	(375 712.5)	R270
LED2	Nabijeni	LED3MM	led	(375 900)	R270
R1	2k2	R0805	rcl	(125 812.5)	R0
R2	300	R1206	rcl	(500 812.5)	R270
R3	300	R1206	rcl	(437.5 587.5)	R180
R4	4k7	R1206	rcl	(1200 87.5)	R180
STABIL	3V3	TO92-ECB	transistor	(375 150)	R180